Simultaneous skeletonization and graph description of airway trees in 3D CT images

Leonardo Flórez-Valencia[1], Alfredo Morales Pinzón[2,3], Jean-Christophe Richard[2], Marcela Hernández Hoyos[3], Maciej Orkisz[2]

[1]Departamento de Ingeniería de Sistemas; Pontificia Universidad Javeriana
Carrera 7 No. 40-62, Bogotá, Colombia

[2]Université de Lyon, CREATIS; CNRS UMR5220; INSERM U1044; INSA-Lyon; Université Lyon 1
Campus LyonTech la Doua, bât. Blaise Pascal, 7 rue Jean Capelle, F-69621 Villeurbanne Cedex, France

[3]Systems and Computing Engineering Department, School of Engineering, Universidad de los Andes
Carrera 1 Este No. 19 A 40, Bogotá, Colombia

`florez-l@javeriana.edu.co, j-christophe.richard@chu-lyon.fr,marc-her@uniandes.edu.co,`
`<alfredo.morales,maciej.orkisz>@creatis.insa-lyon.fr`

Résumé – L'objectif de ce travail est de générer une description symbolique d'arbres bronchiques à partir d'images scanner 3D acquises dans des conditions de ventilation différentes, afin de pouvoir comparer ces arbres entre eux. La méthode s'applique au résultat binaire d'un algorithme de segmentation. Son originalité réside dans une utilisation exhaustive de l'algorithme de Dijkstra et notamment dans une technique originale de détection de points terminaux qui permet d'extraire le squelette et de détecter les bifurcations en une seule passe pendant le parcours inverse du graphe à partir de ces points. Les résultats préliminaires montrent que la méthode proposée est très bien adaptée à l'application visée.

Abstract – The goal of this work is to generate a symbolic description of the airway trees from 3D CT images acquired at different ventilation conditions, in order to compare these trees to each other. The method applies to the binary result of a segmentation algorithm. Its originality resides in a comprehensive use of the Dijkstra's algorithm, namely in a novel technique used to detect the end-points. This technique enables us to extract the skeleton and to detect the bifurcations in a single pass during backtracking from these points. Preliminary results show that the method proposed is well-suited for the intended application.

1 Introduction

The overall goal of this work is to assist physicians at intensive care units by assessing lung aeration in patients with acute respiratory distress syndrome (ARDS). ARDS is a life-threatening state of the lungs (40% mortality), which can be caused by various bacteriological, chemical or mechanical aggressions. Its treatment requires artificial ventilation to support the respiration process. To appropriately set the ventilation parameters (pressure and volume), parenchyma aeration is to be assessed from images acquired at various parameter settings. This needs accurate voxel counting within the lungs, according to standard aeration classes [9]. However, this quantification is hampered by the airways, as these voxels' gray-levels fall within the "over-inflated" class and thus they artificially increase the corresponding score. Therefore, the airways should be removed prior to the quantification, but this cannot be done by simply segmenting out and masking the bronchi in each image. Indeed, the segmentation results from images acquired at distinct pressure and volume settings differ from each other and contain more or less branches, due to such phenomena as pressure-dependent airway diameter and length varia-

tions, partial-volume effect, noise, etc. It is therefore necessary to match the airway trees segmented in each image, in order to find and remove from each image only the common part of the airway tree shared by all the segmentation results. This matching requires a symbolic description of each segmented tree, so that their topology and geometry can be compared.

Obtaining such a symbolic description from segmented binary data is the specific objective of the work herein presented. It usually involves two major steps : (i) skeletonization, and (ii) bifurcation- and end-point detection, in order to build an oriented graph. Skeletonization has been thoroughly studied since many years and has classically been based on morphological erosion with precautions taken to prune spurious branches and avoid shortening the actual branches [3]. Applied in 3D, this approach is time consuming. Though mathematical morphology can be used to detect bifurcations and end-points, a popular solution implemented in the AnalyseSkeleton plugin [1] of ImageJ is based on the Depth-First Search graph-analysis algorithm. A recent approach [8] has used graph theory at both steps. The Dijkstra's algorithm [2] was first applied onto the

---

1. `http://fiji.sc/AnalyzeSkeleton`

segmented image to construct the minimum spanning tree. Then this tree was recursively analyzed to extract the skeleton and infer its symbolic description. Our method follows a similar approach, but fully takes advantage of the minimum spanning tree constructed by the Dijkstra's algorithm, so as to detect the end-points, and to build the symbolic description during backtracking from these points across the tree.

2 Proposed method

As the input required by our method is a binary segmented object $\mathcal{B}$ representing the airways, we selected an existing algorithm devised for such segmentation [7]. Airways in CT images are represented by dark regions typically comprised between -1024 and -600 Hounsfield units, but the actual value of the upper threshold is to be adapted for each image. The algorithm starts from a user-provided seed in the trachea and iteratively performs region growing with a varying threshold, while keeping a voxel count for the region obtained, until "leakage" (i.e., a sudden increase in this count) is detected. The segmentation obtained with the last threshold before this increase is retained. Any algorithm devised to segment the airways in CT images can potentially be used instead of the one we selected.

The skeleton $\mathcal{S}$ of a tree-like structure with tubular branches is as subset of the minimum spanning tree, provided that the latter is constructed using an appropriate cost function, such that the cost is lowest on the tubes' centerlines. The skeleton can be inferred from the minimum spanning tree by connecting the end-point of each branch to the root of the tree. As the CT scans we are interested in are quite big (512^3 voxels, on average), and the whole process (lung segmentation and registration in several images, bronchi segmentation, skeletonization, symbolic description, graph comparison, common tree removal, and final quantification) is to be executed in a short time compatible with decision making at intensive-care units, fast algorithms are needed to perform each step. The Dijkstra's algorithm produces *good* (and in some cases *exact*) results in computing the minimum spanning trees, is fast, and has namely been used in [8] for the purpose of tree skeletonization. While the cost function we use is similar, our methods to detect the end-points and to analyze connections in order to infer a symbolic description are novel, and will be detailed in the next sections.

The whole process can be summarized as follows. The Dijkstra's algorithm is used to build the minimum spanning tree $\mathcal{M}$ with an appropriate cost function (Section 2.1). The nodes of $\mathcal{M}$ are placed in a priority queue ordered in such a way that the node placed on-top is the most likely to be an end-point (Section 2.2). The queue is progressively emptied in a loop by removing, at each step, the node placed on-top and by backtracking from this node towards the tree root (Section 2.3). The nodes of $\mathcal{M}$ visited in the backtracking process are marked. When the backtracking meets an already marked node, this means that a bifurcation point between the currently tracked branch and a previously extracted one has been reached.

2.1 Cost function

Various "medialness" functions have been proposed to assign each voxel a value that increases as the voxel is more likely to be located near a tubular-structure centerline [5]. Similarly to [8] we have chosen to use the distance to the nearest boundary of the segmented object $\mathcal{B}$. Actually, the distance map $D(\mathbf{x})$ (where $\mathbf{x}$ denotes a voxel) is estimated by the Danielsson's algorithm [1], which provides positive values inside $\mathcal{B}$ and negative values outside. The largest values, obtained along the centerlines, can be considered as estimates of the local radius of the tubes. The cost function $c(\mathbf{x})$ is defined as follows :

$$c(\mathbf{x}) = \left\{ \begin{array}{ccc} \infty & ; & D(\mathbf{x}) \leq 0 \\ D^{-2}(\mathbf{x}) & ; & D(\mathbf{x}) > 0. \end{array} \right. \quad (1)$$

After running the Dijkstra's algorithm each voxel $\mathbf{x}$ is seen as a node of thus constructed minimum spanning tree $\mathcal{M}$, and has a corresponding cumulative cost value $C(\mathbf{x})$, which is the sum of the costs $c(\mathbf{x})$ of all the nodes belonging to the minimum-cost path between the root $\mathbf{x}_{root}$ and $\mathbf{x}$.

2.2 End-points detection

To define an "endness" function that takes larger values for nodes more likely to be end-points, we took advantage of the geometry of $\mathcal{B}$ (Fig. 1 left). Candidate end-points are the tips of thin bronchi located far away from the root placed in the trachea. Such points have high accumulated costs $C(\mathbf{x})$ (displayed with hot colors in Figure 1 center) and, as they are thin, their associated distance $D(\mathbf{x})$ is small. Therefore, our "endness" function combines these two values as follows :

$$E(\mathbf{x}) = C(\mathbf{x}) / D^2(\mathbf{x}) \quad (2)$$

The values $E(\mathbf{x})$ are assigned to nodes $\mathbf{x}$ during Dijkstra's algorithm forward propagation, and each node (except the nodes located on the boundary of $\mathcal{B}$, where $D(\mathbf{x}) = 0$) is inserted into a priority queue $\mathcal{Q}$ (actually a max-heap) according to its "endness" $E(\mathbf{x})$. Once the forward propagation is finished, the backtracking starts from the node located on-top of $\mathcal{Q}$, which is very likely to be the tip of the longest and thinnest branch. All the nodes visited during the backtracking are marked and removed from $\mathcal{Q}$, so that in the next pass of the backtracking loop the node on-top of $\mathcal{Q}$ corresponds to the tip of a different branch. The details are given in the next section.

2.3 Symbolic description

Let $pop(\mathcal{Q})$ be the operation of removing the on-top element of $\mathcal{Q}$, and $p(\mathbf{x})$ denote the parent of the node $\mathbf{x}$ in $\mathcal{M}$. For the root $p(\mathbf{x}_{root}) = \mathbf{x}_{root}$. Let us also define the following containers : $\mathcal{L}$ for labels used to mark the visited nodes, $\mathcal{E}$ for end-point coordinates, $\mathcal{Y}$ for bifurcation coordinates, and $\mathcal{S}$ for the skeleton points. Algorithm 1 summarizes our proposal.

Let us note that the nested loop `while` $\mathbf{x} \neq p(\mathbf{x})$, which represents backtracking from an end-point to the root, comprises two modes : (i) as long as a marked node is not encountered, new nodes are marked and added to the skeleton $\mathcal{S}$, then
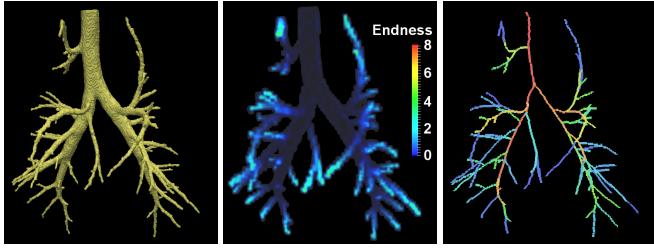
FIGURE 1 – Example (pig A, end-inspiration, highest pressure) of a segmented airway tree (*left*), of the corresponding "end-ness" function represented by the color scale (*center*), and of the final skeleton (148 branches detected) with colors corresponding to different labels, *i.e.*, different branches (*right*).

(ii) after encountering an already marked node, the path towards the root is continued through already visited nodes and their labels are changed (incremented at each bifurcation). In practice, each time a node $\mathbf{x}$ is marked (added to $\mathcal{S}$) or its label is changed, all the nodes in a ball $\mathcal{O}(\mathbf{x}, r)$ centered in $\mathbf{x}$ and with radius $r = 1.5D(\mathbf{x})$ are marked with the same label, so points comprised between the skeleton and the surface of already visited branches cannot be candidate end-points.

3 Results

The method was assessed on 3D CT images from an on-going study of an animal model (pig) with ARDS induced. For each pig, 20 image pairs (end-inspiration/end-expiration) were acquired at various volume/pressure settings. Three pigs were randomly drawn, and for each of them 3 different image pairs were selected, at high, medium, and low pressure, and at constant volume. Thus 18 images with various ventilation conditions were used, containing from 393 (pig C) to 469 (pig A) 512×512-voxel slices with voxel size ranging from $0.46 \times 0.46 \times 0.70$ to $0.58 \times 0.58 \times 1.00 \, \text{mm}^3$. All were segmented using the method described at the beginning of Section 2. Figures 1 (right) and 2 display examples of final results.

As the existing methods usually perform the skeletonization and symbolic description in two distinct steps, we separately evaluated each of these steps. In the absence of ground truth, skeletons extracted using the popular ITK implementation[2], also available as Skeletonize3D plugin of ImageJ[3], of the method proposed by Lee *et al.* [4], were used as "reference". The skeletons generated by our method were compared to the reference ones by using the Hausdorff distance. This distance ranged from 3.3 to 8.5 voxels with a median value of 5.7.

To separately assess the symbolic description, we took as reference method the one implemented in the AnalyseSkeleton plugin of ImageJ, and applied it onto the skeletons resulting from our method. In thus constructed "reference" trees, the total number of branches was 2566, ranging from 69 (pig A, end-

2. www.insight-journal.org/browse/publication/181
3. fiji.sc/Skeletonize3D

Algorithm 1 Skeletonization and descriptor extraction.

1: **procedure** SKELETON_AND_DESCRIPTORS($\mathcal{Q}$)
2:    **let** $\mathcal{L}, \mathcal{E}, \mathcal{Y}, \mathcal{S}$ **be** empty containers.
3:    $bId \leftarrow 0$ ▷ Initialize a branch identifier
4:    **while** $\mathcal{Q}$ is not empty **do**
5:       $\mathbf{x} \leftarrow pop(\mathcal{Q})$
6:       **if** $\mathbf{x} \notin \mathcal{L}$ **then**   ▷ Is $\mathbf{x}$ already labeled ?
7:          $\mathcal{L}(\mathbf{x}) \leftarrow bId$ **and** add $\mathbf{x}$ to $\mathcal{E}$ and to $\mathcal{S}$
8:          $adding\_new\_points \leftarrow TRUE$
9:          **do**
10:             $\mathbf{x} \leftarrow p(\mathbf{x})$
11:             **if** $adding\_new\_points = TRUE$ **then**
12:                **if** $\mathbf{x} \notin \mathcal{S}$ **then**
13:                   $\mathcal{L}(\mathbf{x}) \leftarrow bId$ **and** add $\mathbf{x}$ to $\mathcal{S}$
14:                **else**
15:                   add $\mathbf{x}$ to $\mathcal{Y}$
16:                   $bId \leftarrow bId + 1$
17:                   $adding\_new\_points \leftarrow FALSE$
18:                **end if**
19:             **else**
20:                **if** $\mathbf{x} \notin \mathcal{Y}$ **then** $\mathcal{L}(\mathbf{x}) \leftarrow bId$
21:                **else** $bId \leftarrow bId + 1$
22:             **end if**
23:          **while** $\mathbf{x} \neq p(\mathbf{x})$
24:       **end if**
25:       $\mathbf{x} \leftarrow p(\mathbf{x})$
26:    **end while**
27:    **return**($\mathcal{L}, \mathcal{E}, \mathcal{Y}, \mathcal{S}$)
28: **end procedure**

---

expiration, low pressure) to 237 (pig B, end-expiration, medium pressure) per dataset, lengths ranging from 0.5 to 86.5 mm. The symbolic description extracted from each dataset was compared to the reference by use of a tree-matching algorithm [6] devised to find the common part shared by two anatomical trees obtained from the same subject in different conditions. The total number of unmatched branches (method + reference) ranged from 0 to 9 (4.2%) per dataset, with a median value of 3 (1.8%). This good agreement was confirmed by visual inspection. The computational time (PC with Intel core i7, 8 cores, 6GB RAM) was between 90 and 131 seconds. Its major part was spent on computing the distance map, while the actual skeletonization + symbolic description ranged from 7 to 30 seconds. The average total processing time, 2 minutes, was 11 times shorter than the average skeletonization time alone (22 minutes) of the reference method.

4 Discussion and conclusions

We proposed a fast method to generate skeletons and symbolic description of tree-like binary objects such as airways segmented out of 3D medical images. Our method performs both tasks simultaneously by fully exploiting the minimum spanning tree structure built by the Dijkstra's algorithm, as well as
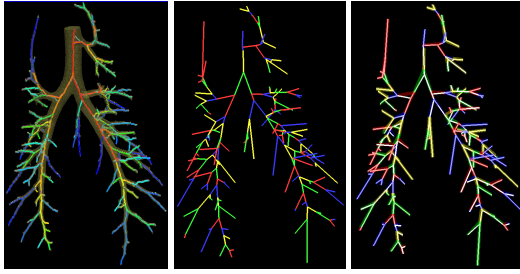
FIGURE 2 – Example of results (pig C, end-inspiration, medium pressure, 224 branches detected) : labeled skeleton displayed within translucent surface of the segmented airways, colors corresponding to different branches (*left*), corresponding graph (*center*), and the same graph superimposed onto the reference in white (*right*) ; here 8 branches remained unmatched.
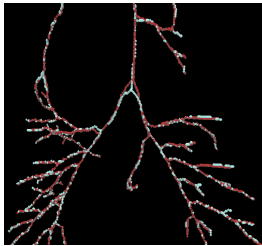

FIGURE 3 – Extracted (red) and reference (light blue) skeletons superimposed. The points generally coincide, except on some bifurcations and branch tips.

the associated costs. It was evaluated on representative images from the intended medical application. In the absence of ground truth, we used as reference popular methods available online, which are significantly slower. The assessment of the skeletons obtained from our method was based of the Hausdorff distance, which detects the worst case, i.e., the largest distance from the object's points to the reference. By visual inspection, the voxels most distant from the reference were located at the tips of some branches and near some bifurcations (Fig. 3). Otherwise, the skeletons were very well aligned with the reference. The symbolic descriptions of these skeletons, generated by the proposed and reference methods, were very similar. Typically, they differed by 2 or 3 branches, which represents less than 2% of the average branch number per dataset.

Overall, the proposed method provided results very close to well-established ones used here as reference, while being much faster. We conclude that it can be successfully used for the intended application. Our method has an additional advantage very useful for this application : the labels assigned to all voxels of the binary object (not only to the skeleton points) in the backtracking step, can be directly exploited to reconstruct and display only a selected subset of branches. In our case, this subset is the common part between airway trees segmented from different ventilation conditions. Let us note that the segmentation method used to generate the input for our method was just an example, and may need to be replaced by a more reliable algorithm. Indeed, if leakage is detected too late, the object segmented may contain a spurious highly irregular part generating numerous branches in the skeleton. Although such branches are rather unlikely to be matched with the tree obtained in different conditions, their extraction and attempts to match them will lead to useless computational-time increase.

Acknowledgments

Références

[1] PE Danielsson. Euclidean distance mapping. *Comput Graphics Image Proc*, 14(3) :227–248, 1980.

[2] EW Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1 :269–271, 1959.

[3] RC González and RE Woods. *Digital Image Processing*, chapter 8, pages 491–494. Addison Wesley, 1993.

[4] TC Lee, RL Kashyap, and CN Chu. Building skeleton models via 3-D medial surface/axis thinning algorithms. *Comput Vision Graph*, 56(6) :462–478, 1994.

[5] D Lesage, ED Angelini, I Bloch, and G Funka-Lea. A review of 3D vessel lumen segmentation techniques : models, features and extraction schemes. *Med Image Anal*, 13(6) :819–845, 2009.

[6] A Morales Pinzón, M Hernández Hoyos, JC Richard, L Flórez-Valencia, and M Orkisz. A tree-matching algorithm : Application to airways in CT images of subjects with ARDS. *Med Image Anal*, submitted, 2015.

[7] K Mori, J Hasegawa, J Toriwaki, H Anno, and K Katada. Recognition of bronchus in three-dimensional X-ray CT images with applications to virtualized bronchoscopy system. In *Proc 13th Int Conf Pattern Recogn*, volume 3, pages 528–532, 1996.

[8] L Verscheure, L Peyrodie, AS Dewalle, N Reyns, N Betrouni, S Mordon, and M Vermandel. Three-dimensional skeletonization and symbolic description in vascular imaging : preliminary results. *Int J Comput Assist Radiol Surg*, 8(2) :233–246, 2013.

[9] SRR Vieira, L Puybasset, J Richecoeur, Q Lu, P Cluzel, PB Gusman, P Coriat, and J Rouby. A lung computed tomographic assessment of positive end-expiratory pressure-induced lung overdistension. *Am J Respir Crit Care Med*, 158(5) :1571–1577, 1998.