Robust Region Growing Based Intrathoracic Airway Tree Segmentation

Rômulo Pinho, Sten Luyckx, and Jan Sijbers

University of Antwerp, Physics Department, VisionLab, Belgium
{romulo.pinho; jan.sijbers}@ua.ac.be
{sten.luyckx}@student.ua.ac.be

Abstract. In this paper, we present a semi-automatic region growing algorithm to segment the intrathoracic airway tree from 3-d CT images. A common problem with region growing is leakage. In order to limit leakage, our method bounds the segmentation using cylinders of adaptive orientation and dimensions. The leaks are detected based on anatomical information of the airways and an algorithm to avoid them is proposed. We also present an algorithm to automatically select a seed point for the segmentation. The method was tested on a dataset of 40 patients and results were quantitatively evaluated based on ground truth data.

1 Introduction

Airway tree segmentation is the process of identifying and extracting from volumetric medical images the structures of the respiratory system that lead the air into the lungs. With the result of the segmentation, doctors and researchers can make measurements, check for abnormalities and generally be assisted in diagnosing diseases in the respiratory system.

In this work, we concentrate on the segmentation of the lower airway tree, namely the trachea, bronchi and bronchioli. Due to the natural complexity of the airways, with several branching levels, and noise or other artefacts present in the image, the segmentation is far from trivial. A common method to solve the problem is region growing [1], and semi- and fully-automated region growing algorithms have been used to segment the airways [2–4]. In this process, the user provides one or more seed points inside the airway structure. From these points, a region is grown by recursively aggregating voxels that pass a certain test of similarity. Common similarity tests check differences in intensity between neighbouring voxels.

One common problem of region growing algorithms is leakage. In the case of the airway tree segmentation, a thin wall separates the structure from neighbouring organs and air inside the lungs. Noise or other artefacts can create holes in this wall and, since the airway lumen and the lung interior have similar voxel intensities, the entire lung can be aggregated to the region. Another problem specific to airway tree segmentation is the early collapse of branches. In this case, the growing process stops too early, resulting in only partially segmented branches.

Tschirren *et al.* proposed an algorithm that takes advantage of the fact that the airway tree is a hierarchical combination of cylindrically shaped objects [5]. In their algorithm, cylinders of adaptive radius and orientation bound the segmentation, facilitating the process of leak detection. Later, Pinho *et al.* proposed improvements to [5] and introduced new ways of detecting leaks using anatomical, instead of pure image features [6]. In the present work, we build upon [6] and add the following contributions:

- propose a heuristic algorithm to automatically select a seed point inside the trachea, since chest CT scans often include the upper airways and other regions, complicating this task;
- use cylinders of adaptive height as well as adaptive radius in order to bound the segmentation;
- propose a new strategy to avoid leaks, by taking into account the fact that they grow through small holes on the edges of the structure being segmented.

The proposed method was evaluated with a dataset of 40 patients, subdivided into training and testing groups. Measures of number of branches, airway tree length, and leakage were taken in order to evaluate the method, by comparing it to ground truth data.

2 Method

We begin with a review of the method proposed in [5] and the ideas introduced in [6]. In [5], multiseeded fuzzy connectivity (MFC) [7] was used to segment the airways' walls and lumen in an iterative process which places adaptive cylinders (or ROIs) around the region to be grown. Airway walls and lumen compete for voxels based on an affinity value $\psi \in [0, 1]$ assigned as a function of voxel intensities. The ROIs bound the region growing and set limits to possible leaks. A leak detector assumes that leaks have a "spongy" structure, with many holes and tunnels. Once a leak is detected with a special morphological operator, the algorithm goes back to the previous step and repeats the segmentation, using what the authors called "directional affinity". This strategy avoids new leaks by assigning affinity values as a function of the intensity of a voxel and of its neighbours lying in the direction of the ROI. Airway branching is detected by computing the skeleton of a region within an ROI, using distance transforms. The branches of the skeleton, their spatial orientation, and the intersections between the region and the borders of the ROI determine the radius and orientation of the ROIs of the next step. The heights of ROIs may change if the segmentation stops exactly at a branching point. This process continues until no more voxels are aggregated.

In [6], the authors proposed improvements to the above algorithm. First, execution speed was increased with a simplification of the skeleton computation: instead of computing the real skeleton, an approximation was obtained by directly linking the centres of gravity of intersections between a region and their respective ROI. For intersections occurring on the side and upper borders of

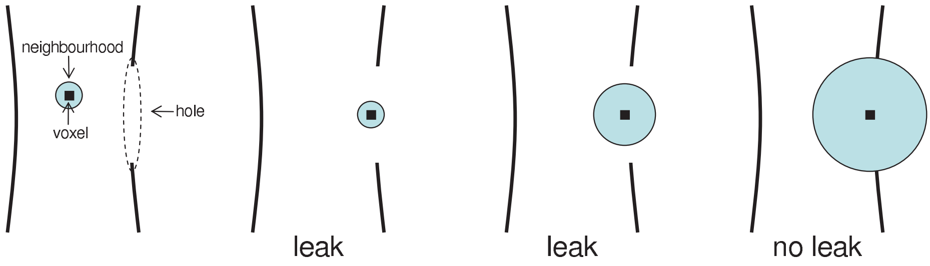

Fig. 1. Avoiding leaks. The segmentation is repeated with an increasing neighbourhood mask until no leaks are detected.

the ROIs, a global centre of gravity for the ROI was obtained and connected to the regions of intersection. This approximation is certainly not precise enough with respect to skeleton accuracy, but suffices for the estimation of the radii and orientations of the ROIs of the next step.

The second improvement dealt with the detection of leaks. Instead of the purely image based approach adopted in [5], the leak detection uses anatomical knowledge about the airways. For instance, the number of offspring branches from one level to the next is usually not larger than 3 or 4 and the radius of a branch is normally a decreasing function of its length and branching level. By checking the number of branches and their areas resulting from intersections with ROIs, leaks can be easily detected. Similar ideas were used in [8], with wavefront propagation algorithms, and in [9], with region growing.

In order to further improve [6] by detecting more airway branches and reaching deeper into the lungs, we propose to extend the region growing with two new ideas. The first is to use ROIs of adaptive height instead of only adaptive radius, such that $h_l = H\rho^{(l-1)}$ and $h_l \geq H_{min}$, where $l \geq 1$ is the current branching level, h_l is the height of the associated cylinder, in millimetres, H is the default initial height, $\rho \in [0, 1]$ is the height change ratio, and H_{min} is the minimum height limit. By doing this, short branches at higher branching levels, which otherwise would not intersect an ROI, can be detected.

Secondly, whenever the segmentation within an ROI is repeated due to a leak, each candidate voxel and its neighbourhood within a mask are analysed. Only if the voxel and all its non-visited neighbours pass the similarity test (voxel intensity lower than a threshold T in our case) is the voxel aggregated to the region. Each time the segmentation is repeated, a mask of higher radius is used, until no leaks are detected or a maximum number of attempts is reached. The reasoning behind this approach is that a leak always occurs due to the presence of holes on the boundaries of the region being segmented, so we are basically trying to discover their sizes. In contrast to the direction affinity adopted in [5], our neighbour affinity technique allows more possibilities of continuing with the segmentation, while trying to avoid leaks. The 2-d scheme of Fig. 1 illustrates the idea and **Algorithm 1** details the process.

Algorithm 1 region_grow(image, seed)

```
 1: /* intialization and computation of 1st ROI */
 2: while ¬ empty(roi_queue) do
 3:    r ← pop(roi_queue)
 4:    roi_region_grow(image, r)
 5:    has_leak ← detect_leak(r) {using anatomical information}
 6:    if has_leak ∧ (count_leak < max_count_leak) then
 7:       set_neighbour_affinity(mask[count_leak]) {set mask to avoid leak}
 8:       count_leak ← count_leak + 1
 9:       reset(image, r)
10:       push(roi_queue, r)
11:    else
12:       if has_leak then
13:          remove_leaking_branches(r) {may remove all branches}
14:       end if
15:       roi_list ← process_roi(r)
16:       for all r_i ∈ roi_list do
17:          push(roi_queue, r_i)
18:       end for
19:       count_leak ← 0
20:       set_neighbour_affinity(null) {switch neighbour affinity off}
21:    end if
22: end while
```

In the algorithm, Step 1 comprises a sequence of steps to compute the first ROI, using the given image and seed point, and push it onto an ROI-queue. After growing a region within the ROI at the front of the queue, Step 5 detects leaks using anatomical information. A leak is identified when the ROI splits into more than 5 regions for levels 1 through 4 and into more than 3 regions for higher levels. In addition, a leak is also identified when the area from a branch to its children increases by a factor $f \geq 2$. The regions of intersection between the ROI and the region grown corresponding to leaks are put in a list. If a leak is detected, Step 7 switches neighbour affinity on by providing the next neighbourhood mask to be used in the similarity tests. The ROI is reset and pushed back onto the queue in Steps 9 and 10, respectively. If leaks are still present after trying all masks, Step 13 removes from the list obtained in Step 5 the corresponding branches. Step 15 processes the remaining branches of the current ROI and returns a list of ROIs for the next iteration, which are pushed onto the queue. Finally, Step 20 switches neighbour affinity off and the process restarts.

Seed Point Selection As mentioned previously, region growing algorithms need one or more seed points to mark the start of the segmentation. Algorithms that automate the seed point selection for the segmentation of the airways typically detect a circular region near the centre of a slice of the image volume. The region is supposed to correspond to the trachea, and the seed point is taken as

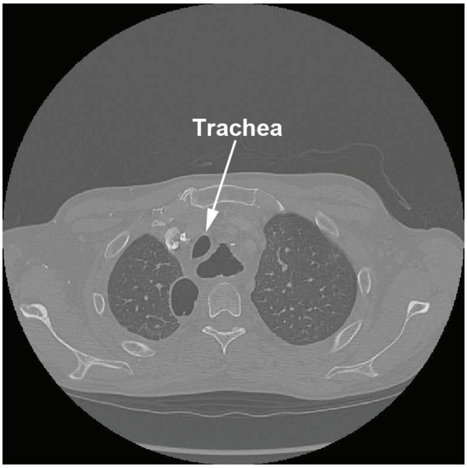

Fig. 2. A situation where other structures may mislead the detection of the trachea in an axial slice of the CT scan.

its centre of gravity. This process may fail if the chosen slice contains misleading regions (e.g., if the CT scan contains parts of the upper airways) or does not contain the trachea at all. Fig. 2 illustrates the former case with a slice containing the trachea, the oesophagus, and a tumour. In the present work, we propose a more robust method to automatically select a seed point inside the trachea.

For one axial slice $i = 1 \ldots N$ of the image volume, the method works as in **Algorithm 2**. Let us use the *threshold below* operation to turn all voxels with intensity below a certain threshold to white and the rest to black [10]. Step 1 thus finds the best threshold to segment the air in the image, which includes the areas inside the lungs and airway lumen, using Otsu's method [11]. Step 2 applies a masked, morphological closing operation to the slice in order to fill all holes. Step 3 identifies 8-connected regions in the resulting image and labels them. Step 6 removes noise, i.e., all regions with size $s \leq S_{min}$ pixels. Step 10 takes care of eliminating narrowed regions, i.e., with excentricity $e > e_{max}$, and steps 12 through 15 identify the region of the slice with highest excentricity, R_{e_i}.

After these steps, a number of regions may be left in each slice. These regions comprise the trachea and areas corresponding to air outside the lungs, the upper airways, the lungs, etc. The challenge is then to choose the slice containing only the trachea or at least to correctly identify it when other structures are present. For this, we minimise a function of several parameters, in order to favour:

- slices with fewer regions, since, in general, the upper part of the trachea tends to appear alone in the image;
- slices that maximise e, of R_{e_i}, given that the upper part of the trachea, just below the infraglotic cavity, tends to be elliptical;
- slices in which the major axis of R_{e_i} is aligned with the sagittal plane;

Algorithm 2 find_trachea(slice$_i$)

threshold(slice$_i$) {segment air}
close(slice$_i$)
regions$_i \leftarrow$ label(slice$_i$)
for all $\mathrm{R}_j \in$ regions$_i$ **do**
  if size(R_j) $< S_{min}$ **then**
    remove(regions$_i$, R_j) {remove noise}
  else
    $e \leftarrow$ excentricity(R_j)
    if $e > e_{max}$ **then**
      remove(regions$_i$, R_j) {remove narrow regions}
    else
      if $e > max_e$ **then**
        $max_e \leftarrow e$
        $\mathrm{R}_{e_i} \leftarrow \mathrm{R}_j$
      end if
    end if
  end if
end for

- slices with lower indexes, since the search is for the top of the trachea (assuming slice 0 coincides with the top position of the CT scan);
- slices in which R_{e_i} is small, which avoids confusion with the lungs;
- slices in which R_{e_i} maximises the area of the ellipse, so that only "regularly" shaped ellipses are chosen.

We therefore define the minimisation as

$$\arg\min_{i=1..N} f(i, e_i, a_i, s_i, r_i) = n_{r_i}\left(\frac{i}{N} + \frac{s_i}{S} + (1 - e_i) + a_i + (1 - r_i)\right), \quad (1)$$

where $\{e_i, a_i, r_i \in [0,1]\}$. In this equation, i is the slice index, $n_{r_i} \geq 1$ is the number of regions of the slice, s_i is the size of R_{e_i} in pixels, with S being a maximum size threshold, e_i is the excentricity of R_{e_i}, a_i is the angle between R_{e_i}'s major axis and the sagittal direction, and r_i is a measure of area maximisation. The latter is computed by taking the ratio between the number of pixels of R_{e_i} and the area of its corresponding ellipse. Lastly, the selected seed point is the centre of gravity of the R_{e_i} that minimises $f(\cdot)$.

3 Experiments

As stated in Section 1, the proposed method was tested with a dataset of 40 patients, provided as part of the workshop and airway segmentation challenge *EXACT09: Extraction of Airways from CT 2009*. The data was subdivided into one training and one testing group, each with 20 patients, numbered CASE01...CASE20 and CASE21...CASE40, respectively. The segmentation was evaluated by a team of trained observers. The aim of the workshop was to compare the performance of different algorithms. For this purpose, a ground truth was constructed from all submitted segmentations and all submissions were evaluated with respect to this ground truth.

The objective of the experiments was to check, for the testing group, how many branches were detected, the segmented tree length and the amount of leakage. The following measures were used to compare the submitted results:

- Branch count: number of branches detected.
- Branch detected: the fraction of branches that were detected with respect to the branches present in the ground truth.
- Tree length: the sum of the length of the centre lines of all correctly detected branches.
- Tree length detected: the fraction of tree length that was detected correctly, relative to the tree length of the ground truth.
- Leakage count: the number of unconnected groups of "correct" regions that are neighbours of a "wrong" region.
- Leakage volume: the volume of regions that are wrongly detected.
- False positive rate: the fraction of the volume of regions that are detected wrongly relative to the volume of all detected regions.

The trachea was excluded from the branch length and branch count related measurements. For the voxel based measures of leakage, both trachea and main bronchi were excluded. Furthermore, the exact airway shape and dimensions were not taken into account.

We implemented algorithms 1 and 2 in C++, and the programs were executed on an Intel® Core™ 2 Quad CPU, at 2.4 GHz, with 8GB of RAM, running under Windows Vista™ Ultimate 64-bits. The region growing algorithm used a single threshold value $T = -800HU$ for the whole airway tree and did not employ multiseeded connectivity, as opposed to [5], since we only segmented the airway lumen, not the walls. In addition, intensities of candidate voxels were averaged within a 6-connected neighbourhood to reduce noise artefacts. The parameters ρ and H_{min} were primarily chosen empirically for the training group, but adjustments were necessary during the experiments with the test set. Eventually, $\rho = 0.85$ provided the best results except for CASE32, for which it was set to $\rho = 0.75$, with $H_{min} = 2$mm in all cases. For the neighbour affinity, we used spherical and cubic masks with radii from 1 to 7 voxels, defining, in this order, 6, 18, 26, 92, 124, 342, 728, 1330, and 2197-neighbourhoods. These masks remained unchanged during the experiments with the test set, but new masks were added until the results for the training group were, at least visually, acceptable. With respect to the seed point selection, we used in **Algorithm 2** $S_{min} = 250$ pixels and $e_{max} = 0.75$, and, in Eq. (1), S was equal to the number of pixels of the slice and $N = \min(300, N_a)$ slices, where N_a is the number of axial slices of the image volume. Again, these values were empirically chosen for the training group, but remained unchanged with the test set.

3.1 Results

The results obtained with the region growing algorithm applied to the testing group[1] can be seen in Table 1. The main difficulty in the segmentation of the airways is to find the balance between the number of segments detected and leakage. In general, it is very difficult to increase the former without allowing the latter to increase as well. Our approach thus remained on the conservative side in terms of branch count and reach, but mostly with low leakage count.

Table 1. Evaluation measures for the twenty cases in the test set.

	Branch count	Branch detected (%)	Tree length (cm)	Tree length detected (%)	Leakage count	Leakage volume (mm^3)	False positive rate (%)
CASE21	69	34.7	39.4	35.7	0	0.0	0.00
CASE22	132	34.1	86.4	26.1	7	160.0	1.14
CASE23	89	31.3	56.6	21.7	6	56.1	0.52
CASE24	69	37.1	56.0	34.4	14	277.2	1.66
CASE25	76	32.5	58.5	23.2	8	557.8	3.27
CASE26	35	43.8	24.3	37.0	0	0.0	0.00
CASE27	36	35.6	25.9	31.9	0	0.0	0.00
CASE28	53	43.1	35.2	32.1	1	473.7	7.60
CASE29	73	39.7	46.9	34.0	4	27.6	0.40
CASE30	47	24.1	33.2	21.7	0	0.0	0.00
CASE31	61	28.5	39.0	22.2	7	578.0	6.98
CASE32	64	27.5	46.6	21.4	2	1740.7	14.34
CASE33	70	41.7	50.2	34.2	5	670.3	11.25
CASE34	140	30.6	85.4	23.9	10	2407.9	12.70
CASE35	95	27.6	61.1	19.8	3	39.7	0.32
CASE36	83	22.8	69.7	16.9	0	0.0	0.00
CASE37	67	36.2	52.3	29.4	2	105.9	1.11
CASE38	28	28.6	23.5	35.3	0	0.0	0.00
CASE39	109	21.0	84.8	20.7	2	93.5	1.04
CASE40	88	22.6	63.6	16.4	13	1420.6	10.24
Mean	74.2	32.1	51.9	26.9	4.2	430.4	3.63
Std. dev.	29.5	6.9	19.6	6.9	4.4	672.3	4.92
Min	28	21.0	23.5	16.4	0	0.0	0.00
1st quartile	53	27.5	35.2	21.4	0	0.0	0.00
Median	70	31.9	51.3	25.0	3	99.7	1.07
3rd quartile	95	39.7	69.7	34.4	8	670.3	10.24
Max	140	43.8	86.4	37.0	14	2407.9	14.34

Further improvements to the proposed method include more robust algorithms to detect leaks and avoid them. One characteristic of the neighbour affinity we adopted is the fact that the resulting segmentation will become thinner as

[1] Provided by the organisers of the workshop.

the neighbourhood mask increases, but this can be corrected with local dilation operations. Another improvement is the use of adaptive intensity thresholds, employed in, e.g., [9, 3]. In fact, we have already tried this approach and produced some primary results. We observed that although the number of branches may increase considerably, so may the number of leaks. As a consequence, this technique must be coupled with efficient leak detection and removal.

With respect to the automatic seed point selection, the proposed algorithm performed very well in all cases. The selected point was always located inside the trachea, at the top. In very few situations, however, the point was set at a lower location. This happened when the trachea had an almost circular shape along all or nearly all of its length. Since we favoured elliptical regions, such a shape happened to appear at slices with higher indexes. Given that the trachea was not considered in these experiments, it was not a problem, but adjustments to the function of Eq. (1) may still be necessary.

Table 2 presents the execution times of the proposed algorithms applied to the testing group. For the region growing, execution time is naturally an increasing function of the number of detected branches, but all executions ran in less than 1 minute, with half of them below 3 seconds and 75% below 8 seconds. The seed point selection, in turn, showed less varying execution times, mostly because $N = 300$ slices for all cases. The differences between cases lay mainly in the complexity of each slice processed by the algorithm. Finally, Fig. 3 presents the segmentation results for 2 patients.

4 Conclusions

In this work, we presented a semi-automatic region growing method for the segmentation of the intrathoracic airways from tomographic scans. The method uses cylinders (or ROIs) of adaptive orientation and dimensions to bound the segmentation. The role of these ROIs is to set a limit to leaks, a common problem with region growing algorithms, and to allow them to be more easily detected. Our approach uses anatomical information about the airways in order to detect the leaks and we proposed a novel algorithm to avoid new leaks once they are detected. We also proposed a heuristic algorithm to automatically select a seed point at the top of the trachea, which is later provided to the region growing algorithm. The method was tested on a dataset of 40 patients, and remained on the conservative side in terms of branch detection, but with a low number of leaks in most cases.

References

1. Gonzalez, R.C., Woods, R.E.: Digital Image Processing. Addison-Wesley Longman Publishing Co., Inc. (2001)
2. Chiplunkar, R., Reinhardt, J.M., Hoffman, E.A.: Segmentation and quantitation of the primary human airway tree. In: Proc. SPIE Medical Imaging. Volume 3033. (1997) 403–414

Table 2. Execution times of the algorithms applied to the test set.

	Region growing (secs)	Seed selection (secs)	Total time (secs)
CASE21	1.98	82.9	84.88
CASE22	6.85	79.6	86.45
CASE23	15.88	64.2	80.08
CASE24	15.02	63.2	78.22
CASE25	31.79	60	91.79
CASE26	1.59	68.8	70.39
CASE27	1.22	57.2	58.42
CASE28	1.83	61	62.83
CASE29	1.67	61.4	63.07
CASE30	0.97	61.2	62.17
CASE31	3.51	64.2	67.71
CASE32	25.74	66.8	92.54
CASE33	5.46	65.6	71.06
CASE34	44.62	57.1	101.72
CASE35	3.6	59.4	63
CASE36	1.62	69	70.62
CASE37	5.3	59.2	64.5
CASE38	2.79	58	60.79
CASE39	1.31	76.5	77.81
CASE40	2.22	70.2	72.42
Mean	8.75	65.27	74.02
Std. dev.	12.08	7.38	12.27
Min	0.97	57.1	58.42
1st quartile	1.66	59.85	63.05
Median	3.15	63.7	70.84
3rd quartile	8.89	68.85	81.28
Max	44.62	82.9	101.72

3. Kiraly, A.P., Higgins, W.E., McLennan, G., Hoffman, E.A., Reinhardt, J.M.: Three-dimensional human airway segmentation methods for clinical virtual bronchoscopy. Academic Radiology **9** (2002) 1153 – 1168
4. Law, T.Y., Heng, P.: Automated extraction of bronchus from 3d CT images of lung based on genetic algorithm and 3d region growing. Volume 3979. (2000) 906–916
5. Tschirren, J., Hoffman, E., McLennan, G., Sonka, M.: Intrathoracic airway trees: segmentation and airway morphology analysis from low-dose CT scans. Medical Imaging, IEEE Transactions on **24** (2005) 1529– 1539
6. Pinho, R., Sijbers, J., Vos, W.: Efficient approaches to intrathoracic airway tree segmentations. In: Proc. IEEE EMBS Benelux Symposium. Volume 2. (2006) 151–154
7. Udupa, J.K., Samarasekera, S.: Fuzzy connectedness and object definition: Theory, algorithms, and applications in image segmentation. Graphical Models and Image Processing **58** (1996) 246 – 261

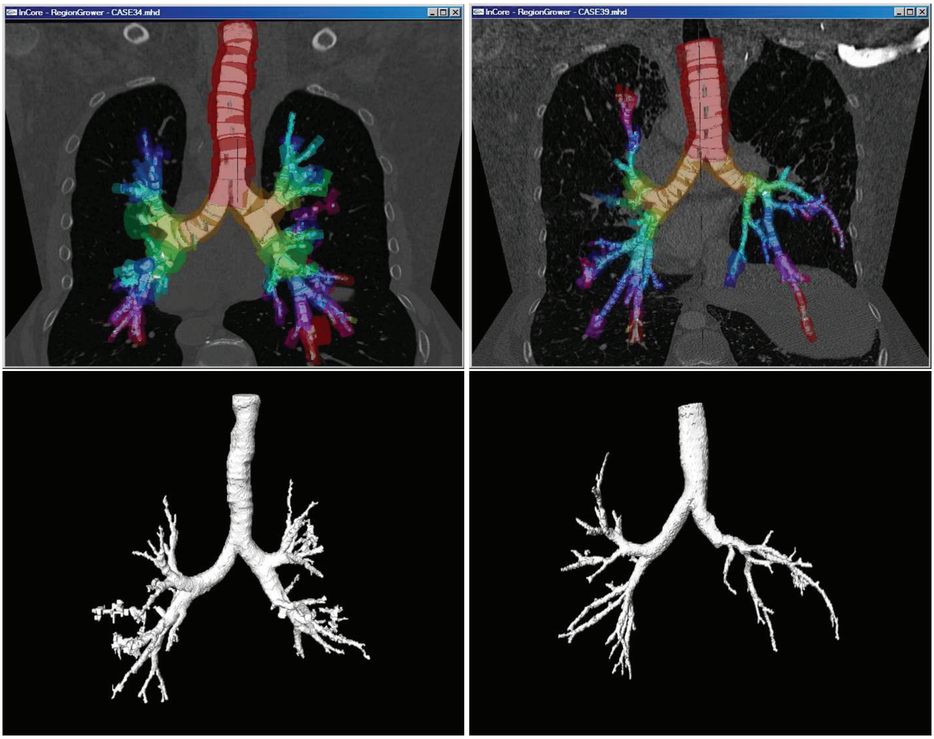

Fig. 3. Segmentations with ROIs coloured per level (T) and respective 3-d reconstructions (B) for CASE34 (L) and CASE39 (R), respectively.

8. Schlathoelter, T., Lorenz, C., Carlsen, I.C., Renisch, S., Deschamps, T.: Simultaneous segmentation and tree reconstruction of the airways for virtual bronchoscopy. In: Proc. SPIE Medical Imaging. Volume 4684. (2002) 103–113
9. van Ginneken, B., Baggerman, W., van Rikxoort, E.M.: Robust segmentation and anatomical labeling of the airway tree from thoracic CT scans. In: MICCAI. (2008) 219–226
10. Shapiro, L.G., Stockman, G.C.: Computer Vision. Prentice Hall (2002)
11. Otsu, N.: A threshold selection method from gray-level histograms. Systems, Man and Cybernetics, IEEE Transactions on **9** (1979) 62–66