Medical Image Analysis 36 (2017) 52–60

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

Improving airway segmentation in computed tomography using leak detection with convolutional networks

Jean-Paul Charbonnier[a,*], Eva M. van Rikxoort[a], Arnaud A.A. Setio[a], Cornelia M. Schaefer-Prokop[c,b], Bram van Ginneken[a], Francesco Ciompi[a]

[a] Diagnostic Image Analysis Group, Radboud University Medical Center, Geert Grooteplein 10, 6525 GA, Nijmegen, The Netherlands
[b] Department of Radiology and Nuclear Medicine, Radboud University Medical Center, Nijmegen, The Netherlands
[c] Department of Radiology, Meander Medisch Centrum, Amersfoort, The Netherlands

ARTICLE INFO

Article history:
Received 6 January 2016
Revised 28 October 2016
Accepted 3 November 2016
Available online 4 November 2016

Keywords:
Airway segmentation
Chest computed tomography
Convolutional networks

ABSTRACT

We propose a novel method to improve airway segmentation in thoracic computed tomography (CT) by detecting and removing leaks. Leak detection is formulated as a classification problem, in which a convolutional network (ConvNet) is trained in a supervised fashion to perform the classification task. In order to increase the segmented airway tree length, we take advantage of the fact that multiple segmentations can be extracted from a given airway segmentation algorithm by varying the parameters that influence the tree length and the amount of leaks. We propose a strategy in which the combination of these segmentations after removing leaks can increase the airway tree length while limiting the amount of leaks. This strategy therefore largely circumvents the need for parameter fine-tuning of a given airway segmentation algorithm.

The ConvNet was trained and evaluated using a subset of inspiratory thoracic CT scans taken from the COPDGene study. Our method was validated on a separate independent set of the EXACT'09 challenge. We show that our method significantly improves the quality of a given leaky airway segmentation, achieving a higher sensitivity at a low false-positive rate compared to all the state-of-the-art methods that entered in EXACT09, and approaching the performance of the combination of all of them.

1. Introduction

Airway tree segmentation in thoracic computed tomography (CT) plays an important role in the analysis of pulmonary diseases. A robust method to automatically segment an airway tree in CT is especially relevant for quantifying airway changes such as bronchial wall thickening, changes in lumen diameter, and pruning of airways. Quantifying these airway changes may be key for improving the diagnosis and treatment planning for pulmonary diseases involving airway pathology such as chronic obstructive pulmonary disease (COPD), cystic fibrosis, or interstitial lung diseases (Pu et al., 2012). In addition, segmented airways can aid in segmenting other pulmonary structures such as lobes, segments, and pulmonary arteries and veins (Bülow et al., 2005; van Rikxoort et al., 2010; Ukil and Reinhardt, 2005; Zhou et al., 2006).

In recent years, two reviews have been published on automated airway analysis (Pu et al., 2012; van Rikxoort and van Ginneken, 2013), providing an extensive overview of different airway segmentation methods. A substantial number of these methods are initialized by performing basic operations on the voxel intensities, such as thresholding or region growing (e.g. as in Fabijasńka (2009), Graham et al. (2010), and van Rikxoort et al. (2009)). These kinds of algorithms rely on the attenuation differences between the lumen, bronchial wall, and surrounding lung parenchyma. However, this difference becomes less pronounced for smaller bronchi because of limitations in resolution and partial volume effects, which often results in a segmentation that leaks into the lung parenchyma. Additional algorithms have been proposed that either do not use region growing or help to control the number of leaks in the segmentation, including algorithm based on morphology or geometry (e.g. Fabijasńka (2009), Fetita et al. (2004), and Graham et al. (2010)), machine learning (e.g. Kitasaka et al. (2010), Lo et al. (2010), and Breitenreicher et al. (2013)), and template matching (e.g. van Rikxoort et al. (2009)). These algorithms usually rely on the appearance of airways on CT (i.e. dark ellipse in 2D, or dark tube in 3D) or on some predefined anatomically based rules to

* Corresponding author.
E-mail address: jean-paul.charbonnier@radboudumc.nl (J.-P. Charbonnier).

http://dx.doi.org/10.1016/j.media.2016.11.001
1361-8415/

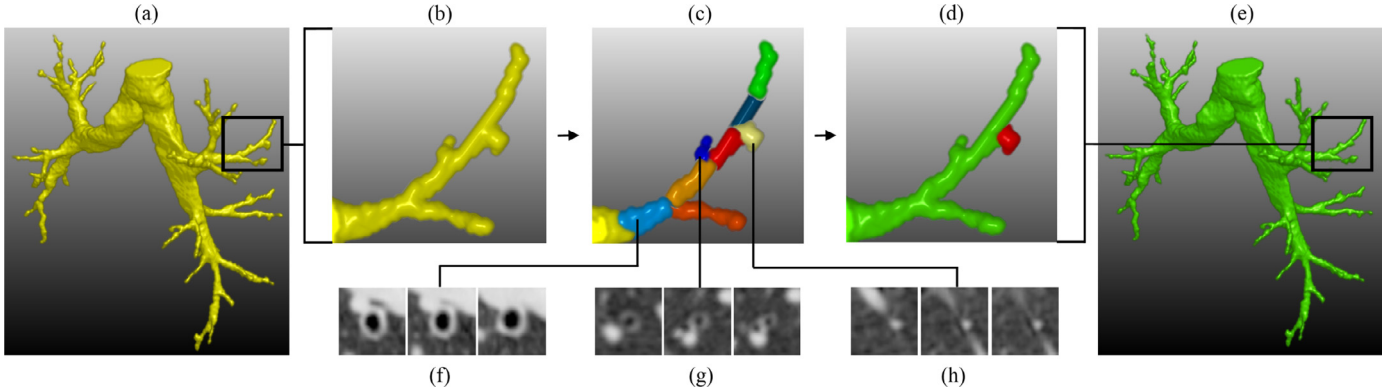

Fig. 1. Schematic overview of the presented method to detect and remove leaks from a given airway segmentation. Given a binary airway segmentation ((a) and (b)), the airways are first subdivided into small airway candidates, represented in different colors in (c) (Section 2.1). From each candidate, a set of three 2D patches is extracted that represents the 3D appearance of the candidate (Section 2.1). An example of three sets of patches is shown in (f), (g), and (h). The ConvNet is used to classify each set of patches as belonging to an airway or to a leak (Section 2.2). In this example, the candidate that corresponds to the set of patches in (h) is classified as leak and is therefore represented in red in (d). The given airway segmentation is improved in (e) by removing the detected leak from the segmentation (Section 3.1). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

restrict the segmentation process. These rules and assumptions, however, may be inaccurate under the presence of noise, artifacts, and diseases.

An airway segmentation challenge was held in 2009 (EXACT'09) (Lo et al., 2012), in which 15 algorithms were evaluated on a heterogeneous set of twenty CT scans. A common issue in all participating methods was how to define a good trade-off between increasing the airway tree length and reducing leaks. In most methods, this trade-off was determined by optimizing the parameters that influence the tree length and the amount of leaks. One of the main conclusions of the challenge was that detecting small bronchi without including leaks in the segmentation is still an unsolved problem. An additional conclusion was that different algorithms provide complementary information, which was shown by the fact that a combination of all participating algorithms provided significantly longer airway trees without increasing amounts of leaks.

Parameter selection is an important step in many airway segmentation algorithms in order to get a decent tree length without leaks (Lo et al., 2012). However, fine-tuning the parameters that influence the tree length and the amount of leaks is often a difficult and tedious task, which may in addition be dependent on the quality of the CT scan. A selected trade-off therefore usually favors limiting the amount of leaks at a cost of detecting fewer smaller bronchi.

We propose a new approach to detect and remove leaks in a given airway segmentation, which conceptually differs from what has been done in the literature. We formulate leak detection as a classification problem and show that leak reduction even allows to increase the detected airway tree length, while keeping the amount of leaks limited. We take advantage of the fact that a range of different segmentations can be extracted from a single airway segmentation algorithm by varying its parameter setting, i.e. the level of restrictions. We propose a strategy to combine these segmentations after leak reduction, which largely circumvents the need for parameter fine-tuning of the given airway segmentation algorithm.

Supervised representation learning techniques have been shown to provide rich descriptions of the data at hand, without the need for engineering application-specific features. In particular, convolutional networks (ConvNets) (LeCun et al., 1998; 2015; Schmidhuber, 2015) have been shown to quickly outperform state of the art approaches for many image classification tasks (e.g. Krizhevsky et al. (2012), Sermanet et al. (2014), Simonyan and Zisserman (2014), and Szegedy et al. (2014)). ConvNets typically consist of a stack of several convolutional and pooling layers, followed by a final set of fully-connected layers and, typically, a soft-max layer. One of the key features of ConvNets is that they can be trained end-to-end is a supervised fashion, using raw data as input and the target label as output. Consequently, the parameters of the network, namely the coefficients of the filters used in convolutions and the weights of fully-connected layers, are learned in order to give a rich representation of the data at hand. Thanks to this characteristic, ConvNets have quickly become the state of the art approach in fields of computer vision and speech recognition where a large amount of data is available. ConvNets have not been largely applied yet to the field of medical image analysis, but represent a powerful classification framework that is highly suited for the proposed airway segmentation approach. For this reason, the proposed approach uses ConvNets for detecting leaks as detailed in Section 2.

The method is evaluated using the twenty CT scans of the EXACT09 challenge (Lo et al., 2012), which is an internationally recognized reference standard for airway segmentation. This test set consisted of a completely independent heterogeneous set of scans, including both inspiration and expiration scans with different reconstruction kernels and different severity of interstitial lung diseases.

2. Leak detection using ConvNets

A schematic overview of the proposed method is shown in Fig. 1. A binary airway segmentation is extracted from thoracic CT scans using a given airway segmentation algorithm. In order to detect leaks in this segmentation, we subdivide the segmentation into small sections, referred to as an airway candidate, and extract a set of 2D patches that capture the 3D appearance of each candidate. A ConvNet is used to classify these candidates based on the set of 2D patches. The procedure for leak detection based on candidate detection and classification is detailed in the following sections.

2.1. Airway candidates

Given a binary airway segmentation A_I, our goal is to obtain a representation of the local 3D appearance of short airway candidates c via a set of 2D patches. Candidates are defined by subdividing the segmented airway branches into atomic structures of equal length.

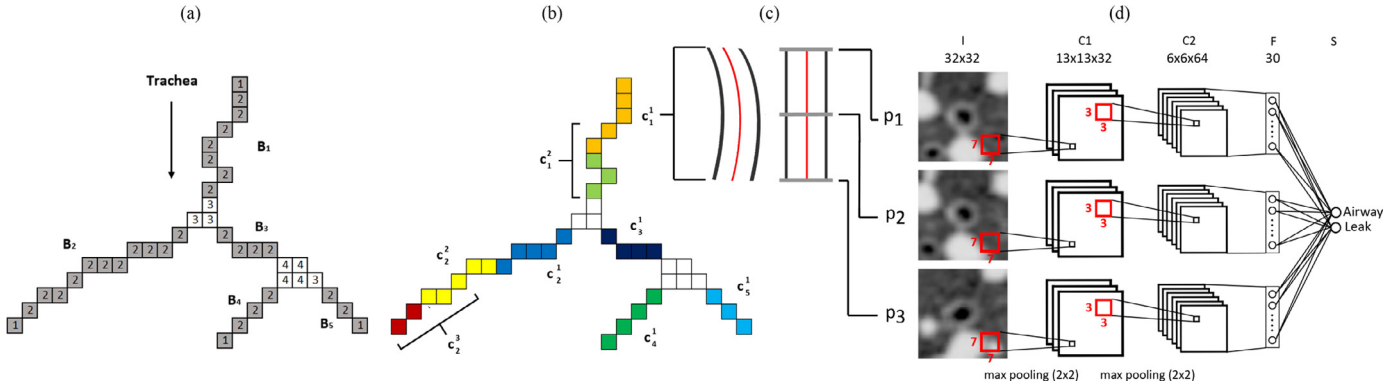

Fig. 2. A detailed overview of leak detection using a convolutional network. (a) 2D example of branch extraction based on centerline voxels x_c, where each x_c includes the number of neighboring voxels $\Omega(x_c)$. Each x_c for which $\Omega(x_c) > 2$ is labeled as belonging to a bifurcation (white voxels), whereas all other voxels are labeled as belong to an airway branch (gray voxels). Airway branches B_i are successively labeled from trachea towards the higher generations. (b) Each B_i is subdivided into a sequence of candidates with length $\mathcal{L}$, where each candidate is represented as a single color. Note that only the last candidate of a branch (e.g. c_1^2 and c_2^3) is allowed to partially overlap. (c) A schematic example of a candidate on which a multi planar reformation is performed that straightens the airway candidate. Three patches (p_1, p_2 and p_3) are extracted orthogonal to the reformatted centerline. (d) The convolutional network used to classify each candidate, where *I* refers to the input layer, *C*1 and *C*2 to the first and second convolutional layers, *F* to the fully connected layer, and *S* to the soft max layer. The three input patches p_1, p_2, and p_3, each have a separate (identical) stack of layers, which are combined in the fully connected layer to perform the classification. The sizes mentioned on top of the layers refers to only a single stack, meaning that the soft max layer gets the input of three fully connected layers, i.e. 3 × 30 filters. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Airway candidate extraction

We introduce an atomic structure that is a candidate for being labeled as leak or airway. These candidates are defined as a small section of an airway branch with a predefined length $\mathcal{L}$. Airway candidates are extracted by applying three sequential steps: (1) skeletonisation of the airway segmentation, (2) airway branches extraction, and (3) subdivision of branches into airway candidates.

Airway centerlines are extracted as a set of connected centerline voxels $X = \{x_c\}$ (Palagyi and Kuba, 1998), where connectivity of centerline voxels is defined in a 26-neighborhood. The number of neighboring centerline voxels of x_c is indicated by $\Omega(x_c)$. All x_c that have more than two neighboring voxels ($\Omega(x_c) > 2$) are labeled as bifurcation voxels, whereas the remaining centerline voxels ($\Omega(x_c) \leq 2$) are labeled as branch voxels. Based on these labels, a branch B_i is given by:

$$B_i = \{x_c : \Omega(x_c) \leq 2, x_c \text{ are connected}\} \quad (1)$$

A schematic example of the construction of branches is shown in Fig. 2a. By considering the trachea as root, the hierarchy of the airway tree is extracted by subsequently labeling each branch relative to the root (Fig. 2a).

Given the airway branches, we extract candidates with the same length. A sequence of candidates $\{c_i^1, c_i^2, \ldots, c_i^{k-1}, c_i^k\}$ is extracted by subdividing a branch B_i in a sequential non-overlapping order, based on the topological direction of the airway tree. A schematic example of this process is shown in Fig. 2. Note that since a branch is not guaranteed to be exactly partitioned into candidates of length $\mathcal{L}$, the last candidate c_i^k in a branch is allowed to partially overlap with its parent. In case a branch is smaller than $\mathcal{L}$, the entire branch is treated as a single candidate. As a final step, an Euclidean distance transform is performed on the centerline in order to assign each voxel $x \in A_I$ to the appropriate candidate.

Airway candidate representation

The 3D appearance of a candidate is captured by extracting three 2D patches p_1, p_2, and p_3 (Fig. 2c and d). A multi planar reformation (MPR) of the original CT image is performed (using trilinear interpolation) that straightens the candidate centerline, which allows to extract patches orthogonal to the direction of the centerline (Fig. 2c). The size of each of the patches was fixed to 15 × 15 mm (32 × 32 pixels), which provided a good trade-off between a detailed view of the smaller bronchi and the possibility to include intermediate and larger bronchi. The extracted patches p_1, p_2, and p_3 corresponded to the beginning (p_1), middle (p_2), and end (p_3) of the reformatted candidate, as schematically illustrated in Fig. 2c.

2.2. Convolutional network

A ConvNet was designed and trained to classify a candidate into airway or leak, where a single candidate is represented by the three extracted 2D patches p_1, p_2, and p_3. Each patch was preprocessed by rescaling the voxel intensities between [0, 1], in order to promote a faster convergence during optimization of the ConvNet (LeCun et al., 1998; Ioffe and Szegedy, 2015). Rescaling was done by clamping the Hounsfield units between a predefined range $[I_{\min}, I_{\max}]$ and dividing the clamped value by the extent of the new range, i.e. $I_{\max} - I_{\min}$. Note that the performed rescaling was done in the exact same way for each patch in order to retain the actual Hounsfield unit information.

The design of the ConvNet was determined by optimizing several parameters that define the architecture of the network, i.e. number of convolutional layers, number of filters per layer, size of filters per layer, dropout, and learning rate, similar to what is done in Setio et al. (2016). The optimization of these parameters was done by training multiple ConvNets with different sets of parameters, where the performance was determined based on the area under the ROC-curve of the validation set. The architecture of the final used ConvNet is outlined in Fig. 2d.

A candidate is classified by processing p_1, p_2, and p_3 in parallel by passing each patch through one of three separate identical stacks of convolutional and max pooling layers. Each stack consists of a first convolutional layer C1 (32 filters of 7 × 7 pixels), a second convolutional layer C2 (64 filters of 3 × 3 pixels), and a fully connected layer F (30 units). Each convolution is performed with a fixed stride of 1 pixel, without performing spatial padding. Max pooling is performed by a 2 × 2 window (with a stride of 2) after each convolutional layer. The fully connected layers of each of the three stacks are combined in the final soft-max layer S (2 units) to perform the classification. Rectified linear activation units (ReLU) (Krizhevsky et al., 2012) were used in all layers. The designed ConvNet was implemented in Theano (Al-Rfou et al., 2016; Bergstra et al., 2010).

In order to train the ConvNet, we defined a training and a validation set that each consisted of candidates labeled as leak (pos-

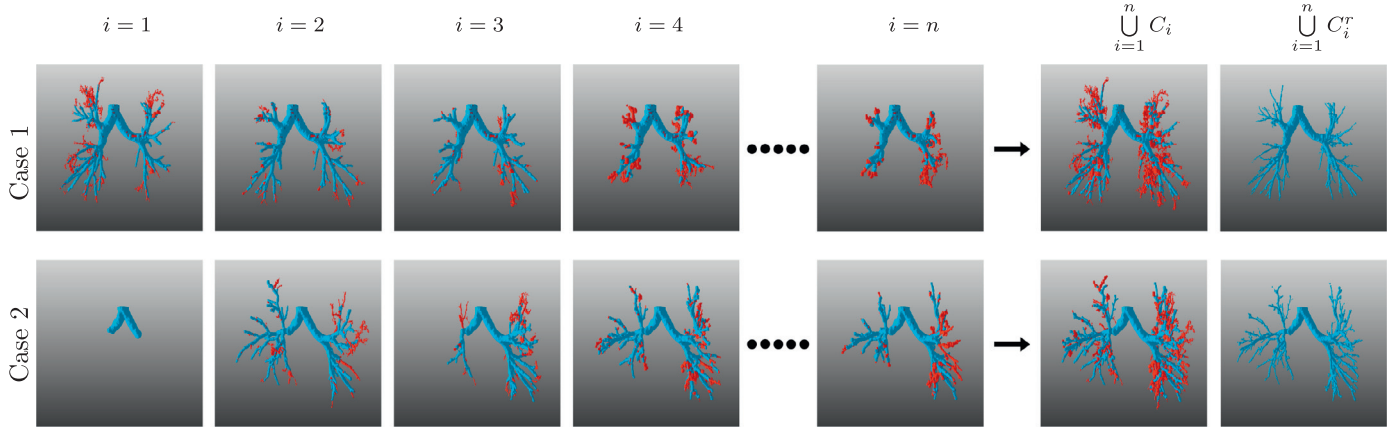

Fig. 3. Two cases are shown to illustrate a practical applications of our leak detection method designed to increase the tree length of an airway segmentation while limiting the amount of leaks (Section 3.2). Coarsely-tuned airway segmentations C_i (indicated in blue and red) are extracted by varying the tunable parameters of a given airway segmentation algorithm, where i indicates a unique combination of tunable parameters. We apply our proposed leak reduction method (Section 3.1) to each C_i in order to extract leak reduced segmentations C_i^r (indicated in blue). $\bigcup_{i=1}^{n} C_i$ indicates the union of all C_i, whereas $\bigcup_{i=1}^{n} C_i^r$ indicates the union of all C_i^r. Note that the same set of parameters in the given airway segmentation algorithm (e.g. $i = 1$) may result in a segmentation with a different trade-off across cases. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

itive class) or airway (negative class). Given the training set, the weights of the ConvNet were learned by minimizing the cross-entropy error between labels and class posterior probabilities via stochastic gradient descent. RMSProp (Tieleman and Hinton, 2012) was used for adapting the learning rates and the weights were updated using mini-batches of 128 samples. A network configuration was selected that maximizes the accuracy on a balanced validation set, where training was stopped after 10 epochs if there was no improvement in accuracy on the validation set.

The number of positive and negative candidates in both the training and validation set were increased using multiple data augmentation techniques. Although augmentation was performed on a patch level, each of the three patches within a candidate was augmented in the same way to preserve the orientation among patches. Patch rotation with angles of 0°, 90°, 180°, and 270° and horizontal flipping were used to obtain eight examples per candidate. In order to better balance the positive and negative candidates, we additionally augmented the under represented class by mirroring the patch order within a candidate in the direction of the airway i.e. $\{p_3, p_2, p_1\}$ instead of $\{p_1, p_2, p_3\}$.

3. Application to improve airway segmentation

In this section we show two practical applications of our leak detection method used to improve a given airway segmentation. Application 1 aims at improving the quality of an existing airway segmentation by removing leaks from the segmentation. Application 2 is used to improve the quality of a given airway segmentation algorithm whose performance depends on a set of tunable parameters. In this application we take advantage of the fact that a range of different segmentations can be extracted from a given airway segmentation algorithm by varying these tunable parameters. We propose a strategy in which a combination of these segmentations after removing leaks can improve the airway segmentation.

3.1. Application 1: removing leaks from segmentation

The first application of our leak detection method consists of reducing the amount of leaks in a given binary airway segmentation A_I. This segmentation is first reduced to a set of candidates, that are classified by the proposed ConvNet to identify leaks and remove them from A_I (Section 2). Removing candidates that are partially leak and partially an airway can result in a set of disconnected airway branches. When the gap between the disconnected airway branches does not exceed a maximum distance of $\mathcal{L}_{\max}$, the airway continuation is restored by reconnecting the disconnected branches using the Euclidean shortest path. The diameter of this reconnection is interpolated from the diameters of the adjacent candidates. Note that the interpolated part of a branch is not allowed to exceed the original segmentation A_I.

3.2. Application 2: combining multiple segmentations

The second application of our leak detection method aims at improving the results of a given airway segmentation algorithm by increasing the detected airway tree length, while keeping the amount of leaks limited. We take advantage of the fact that different segmentations can be extracted from a single airway segmentation algorithm by varying parameters that influence the tree length and the amount of leaks. Instead of optimizing this set of parameters to extract a single finely-tuned segmentation, n coarsely-tuned segmentation C_i are extracted from n combinations of parameters, where i indicates a unique combination of parameters. In Fig. 3, an example of coarsely-tuned segmentations C_i is shown for two patients. Fig. 3 clearly shows that each set of parameters results in a coarsely-tuned segmentation with a different trade-off between tree lengths and leakage counts. In addition, the same set of parameters used for two different patients (e.g. $i = 1$) may result in differently tuned segmentation, which is the reason why fine-tuning of this trade-off is difficult.

In order to construct an improved segmentation, we apply Application 1 to each coarsely-tuned segmentation $C_1, C_2, C_3, C_4, \ldots, C_n$, resulting in n leak reduced segmentations $C_1^r, C_2^r, C_3^r, C_4^r, \ldots, C_n^r$. Since these leak reduced segmentations contain complementary information, their union results in an improved segmentation $A_u^r = \bigcup_{i=1}^{n} C_i^r$ that benefits from a low leakage count and a higher tree length. Fig. 3 visualizes this way of combining leak reduced segmentations. In this figure, the blue segmentations in columns $i = 1, 2, 3, 4$ and n indicate to the reduced segmentations $C_1^r, C_2^r, C_3^r, C_4^r, \ldots, C_n^r$, whereas the red parts of the segmentations are the removed leaks. The two last columns of Fig. 3 show the union of all coarsely-tuned segmentations $A_u = \bigcup_{i=1}^{n} C_i$ and the union of all n leak reduced segmentations $A_u^r = \bigcup_{i=1}^{n} C_i^r$.

4. Data

4.1. COPDGene study

The proposed ConvNet was trained and evaluated on a randomly selected set of 45 high-dose (120 kVp, 200 mAs) full inspiration thoracic CT scans taken from the COPDGene study (Regan et al., 2010). All scans were reconstructed to 512 × 512 matrices, with in-plane voxel sizes between 0.50 and 0.88 mm and slice thickness between 0.63 and 0.90 mm. Liberal airway segmentations were extracted from these scans, containing both airways and a substantial amount of leaks. A total of 32,430 candidates were extracted from the binary segmentations, excluding the trachea and main bronchi. Each candidate was labeled as airway or leak based on the set of 2D patches by an observer specifically trained for this task. The observer was instructed to only label the candidate as airway if the entire candidate (i.e. p_1, p_2, and p_3) was part of an airway, which resulted in 23,814 airways and 8616 leaks. Augmentation of the candidates in the training and validation set was performed as explained in Section 2.2.

4.2. EXACT'09 challenge

The method was evaluated on a completely independent test set of twenty CT scans taken from the internationally recognized EXACT'09 challenge for airway segmentation. This test data represented an heterogeneous set of full inspiration and full expiration scans obtained from subject ranging from healthy volunteers to patients with severe lung disease. The scans were acquired at different sites using several different scanners, scanning protocols, reconstruction parameters, dose (120/140 kVp, 10.0–411.5 mAs), slice thickness (0.50–1.0 mm), and in-plane voxel sizes (0.55–0.78 mm). More information on the specific description of these scans can be found in Lo et al. (2012). Since the EXACT'09 challenge is not maintained at this moment, and new submissions are not processed, we requested the reference standard from the organizers of the challenge and re-implemented the evaluation algorithm (as described in Lo et al. (2012)) in order to compare our results to the other participants of this challenge.

5. Experiments and results

Four experiments were performed to evaluate the performance of our method. In the first experiment, the performance of leak detection with the proposed ConvNet (Section 2) was evaluated on a candidate level using data from the COPDGene study. This proposed approach was compared to two alternative classification method in experiment 2. In the third experiment, we applied leak reduction to given airway segmentations as proposed in Application 1 (Section 3.1), in order to evaluate the ability of our method to reduce leaks in a given airway segmentation. In the fourth experiment, we evaluate the proposed strategy of combining different segmentations extracted from a single given airway segmentation algorithm, as explained in Application 2 (Section 3.2). This combination strategy aims at increasing the airway tree length and keeping leaks limited. Both the third and fourth experiments were performed on the twenty test cases from the EXACT'09 challenge and are meant as an external validation of our method on an heterogeneous data set.

5.1. Experimental setup

For the purposes of this study, we used an average candidate length $\mathcal{L} = 3.5$ mm to be able to classify small parts of a segmentation. The max reconstruction length $\mathcal{L}_{max}$ was chosen to be 40 mm. The rescaling of the patch intensities was done using $I_{min} = -1000$ and $I_{max} = 400$. For training the ConvNet, the maximum validation accuracy was achieved after 6 epochs.

Table 1
An overview of the amount of candidates used to train and test the ConvNet. For training and validation, patch rotation with angles of 0°, 90°, 180°, and 270° and horizontal flipping were used to obtain eight examples per candidate. Since leaks were under represented, additional augmentation was performed on these candidates by mirroring the patch order in the direction of the airway i.e. $\{p_3, p_2, p_1\}$ instead of $\{p_1, p_2, p_3\}$. The number of airway and leak candidates were balanced after augmentation.

	Training	Validation	Testing
Number of scans	27	9	9
Number of candidates before augmentation			
total	19,035	7292	**6103**
airway	13,641	5635	4538
leak	5394	1657	1565
Number of candidates after augmentation			
total	195,432	71,592	–
airway	109,128	45,080	–
leak	86,304	26,512	–
Number of candidates after balancing			
total	**172,608**	**53,024**	–
airway	86,304	26,512	–
leak	86,304	26,512	–

5.2. Experiment 1

In the first experiment, we evaluated the proposed leak detection method (Section 2) on a candidate level using the 45 scans taken from the COPDGene study. Scans were divided into a training, validation, and test set in a 3:1:1 ratio, respectively. An overview of the number of extracted candidates per set before and after augmentation is shown in Table 1. Since both the training and validation set were unbalanced after augmentation (i.e. more airway candidates compared to leaks), we balanced both data sets by randomly selecting airway candidates to match the number of leak candidates. The balanced training set therefore consisted of 172,608 candidates (27 scans) and the validation set of 53,024 candidates (9 scans). Candidates extracted from the remaining 9 scans (i.e. 4538 airways and 1565 leaks) were used for testing. The ROC curve of this experiment is shown in Fig. 4, with an area under the ROC curve (Az) of 0.994. With an operation point of 0.5, a total of 5959 candidates were correctly classified (4443 airways and 1516 leaks), and 180 candidates were assigned to the wrong class (120 false positives and 60 false negatives). This resulted in an accuracy of 0.97, a specificity of 0.97 and a sensitivity of 0.96. Fig. 5 shows examples of patches classified as true positives, true negatives, false positives and false negatives, where the positive class refers to leaks and the negative class to airways. The execution time of our ConvNet for the classification of a single candidate is 0.00198 s on a standard PC with a GPU GeForce GTX TITAN X. Since the airways of one scan typically consist of 500–1000 candidates, execution time of the ConvNet is between 1 and 2 s per scan. The rest of the pipeline (e.g. extracting the candidates and patches, performing augmentation, etc.) takes between 3–5 min per scan.

5.3. Experiment 2

In the second experiment, a comparison was made between our proposed ConvNet and two alternative classification approaches. In the first approach, we transformed the voxel intensities values of the patches of a candidate (i.e. p_1, p_2, p_3) into a single feature vector of $32 \times 32 \times 3 = 3072$ features. In the second additional approach, we used a method based on representation learning as presented by Coates et al. (2011) to extract features and perform the

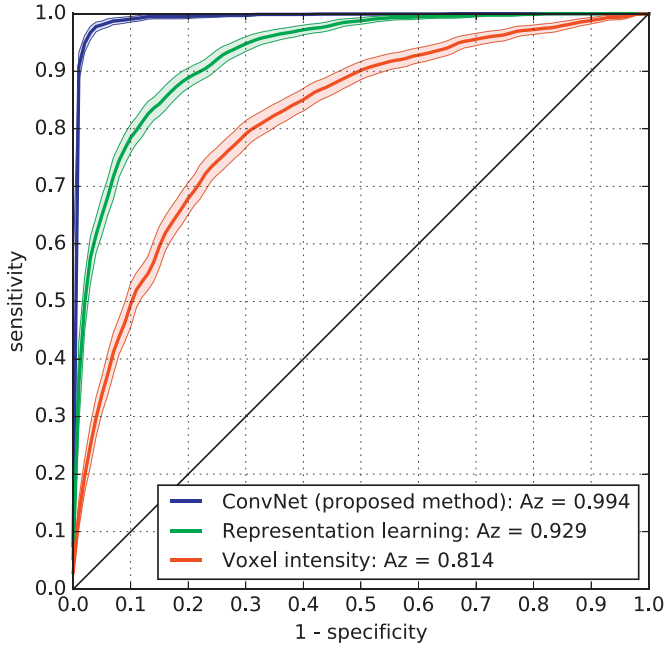

Fig. 4. Result of Experiments 1 and 2 showing the performance on a candidate level of the proposed ConvNet (in blue) and the two alternative classification approaches (i.e. representation learning approach in green and voxel intensity approach in red). The confidence interval is shown as a shades region around the ROC curves. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

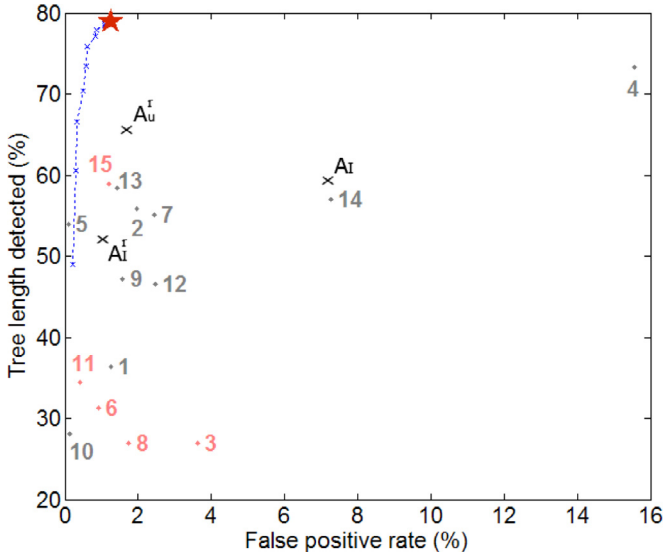

Fig. 6. Figure taken from the EXACT'09 airway segmentation challenge (Lo et al., 2012), which shows the average tree length versus average false positive rate (i.e. leakage percentage) of all 15 participating teams (semi-automatic algorithms are shown in red). A fusion scheme of different combinations of the 15 algorithms are shown as blue stars, where the red star indicates the fusion of all 15 algorithms as proposed in Lo et al. (2012). The results of our method are included as A_I, A_I^r, and A_u^r. A_I refers to the evaluation of algorithm 14 with our reimplemented EXACT'09 evaluation. A_I^r refers to the evaluation performed in Experiment 3, using leak reduction on A_I. A_I^r refers to the evaluation in Experiment 4, where a combination of coarsely-tuned segmentations extracted from algorithm 14 is used to increase the segmented tree length. Note that our method was able to increase the segmented tree length while limiting the amount of leaks. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

classification. Our validation data set was used to learn a set of representative patches in an unsupervised way using a soft version of a K-means clustering algorithm. This set of representative patches was used to extract a feature vector from the input patches p_1, p_2, p_3. The final classification of both additional approaches was performed on the test set using a linear Support Vector Machine classifier, which was the best performing classifier in the method of Coates et al. (2011). The ROC curves of these additional approaches are included in Fig. 4, with an Az of 0.814 for the first approach and an Az of 0.929 for the second approach. Note that both alternative classification approached are outperformed by our ConvNet approach.

5.4. Experiment 3

In the third experiment, we evaluated the ability of our method to reduce leaks in given airway segmentations from the EXACT'09 challenge using Application 1. Fig. 6 was taken from the EXACT'09 challenge, in which the results of all participating algorithms (algorithms 1–15) are summarized. The airway segmentations provided by algorithm 14 (van Rikxoort et al., 2009) were taken as input segmentations A_I to our method. The original evaluation of these segmentations shows that algorithm 14 finds many airways but

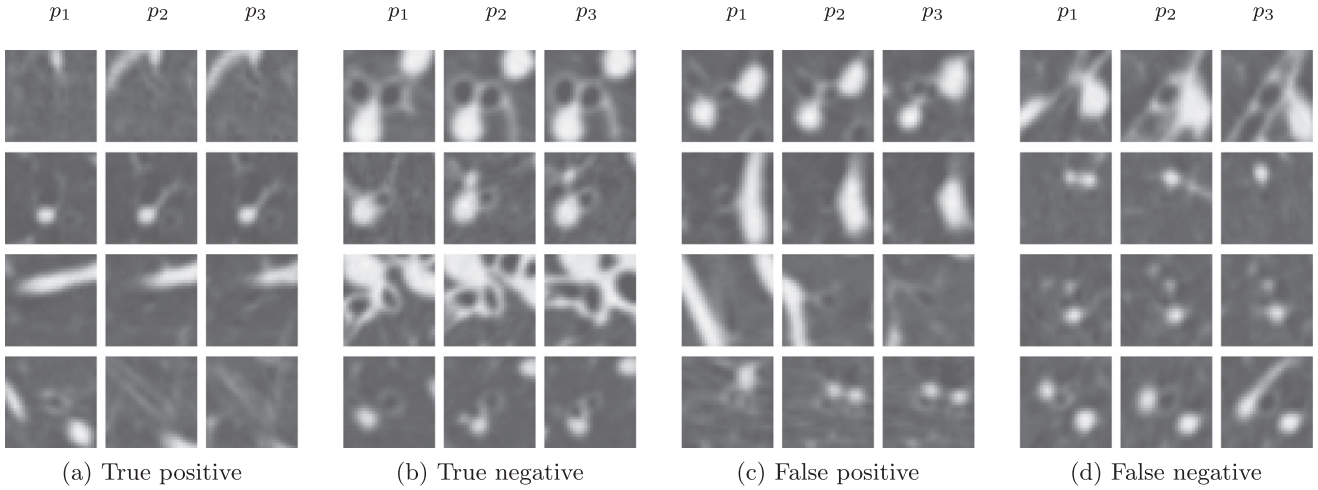

Fig. 5. Example patches that were extracted from the classified candidates in Experiment 1, where the positive class refers to leaks and the negative class to airways. Each row consists of four candidates, where each candidate is represented by the set of three patches (from left to right p_1, p_2, and p_3). A standard lung window level (width = 1600, center = -600) is used to show all patches.

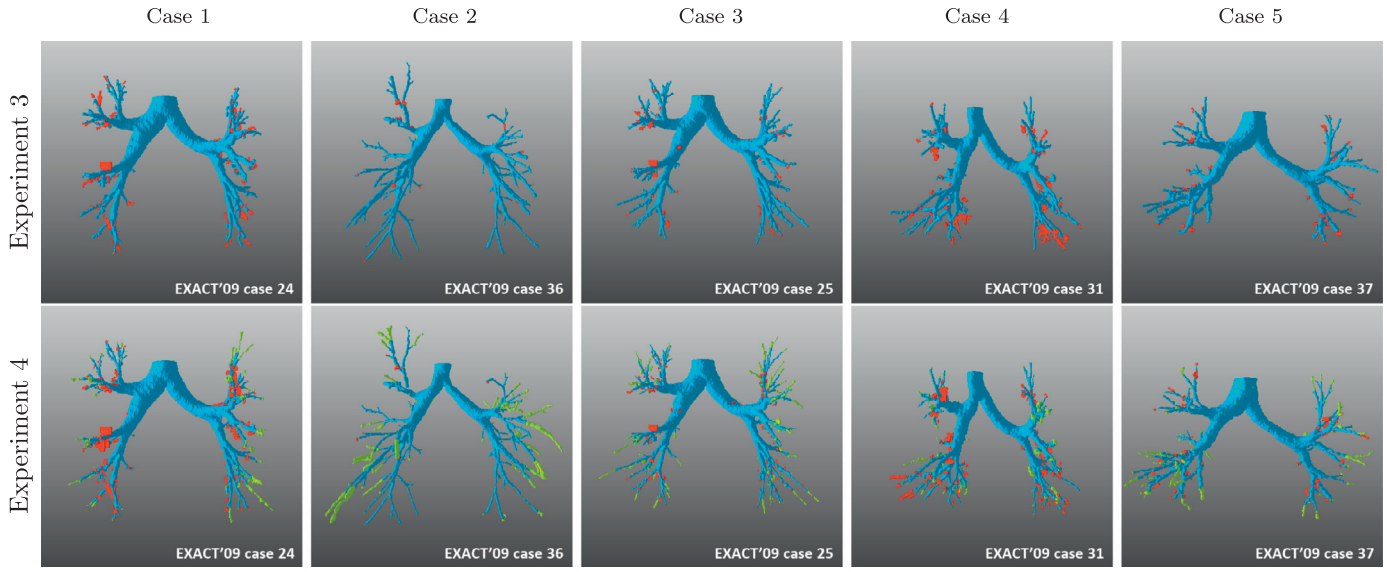

Fig. 7. Results of the third and fourth experiments, showing five cases from the EXACT'09 challenge. In experiment 3 (first row), red indicates voxels that were in the original given segmentations A_I, but classified as leak by our proposed method. The blue parts indicate the improved segmentation A_I^r. In experiment 4 (second row), red indicates voxels that were in the original given segmentations A_I, but not included in the improved segmentations A_u^r. The green parts indicate voxels that were only included in the improved segmentations A_u^r, whereas blue indicates voxels that were both in A^I and A_u^r. The improved segmentations in the first row (A_I^r), were constructed by performing leak reduction on A_I (Section 3.1). The improved segmentations in the second row (A_u^r) were constructed by combining multiple coarsely-tuned segmentations extracted from algorithm 14 of the EXACT'09 challenge (Section 3.2). Each rendering was anterior viewed with a field of view of 250x250mm. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

also produces a substantial amount of leaks, compared to the other participants of the challenge. Since the challenge is not maintained at this moment, we first re-analyzed the original submission of algorithm 14 (indicated as A_I in Fig. 6) in order to validate our reimplementation of the EXACT'09 evaluation. This resulted in a detected tree length of 59.0% and a false positive rate of 7.13%, which is very close to the original result (i.e. a detected tree length of 57.0% and a false positive rate of 7.27%). The small difference between these results was mainly because of differences in the way airway branches were automatically labeled. We therefore consider our re-implementation to be a highly comparable alternative to the original EXACT'09 evaluation.

We applied the proposed leakage reduction method to A_I as explained in Section 3.1 (Application 1), which produced leak reduced segmentations A_I^r. Evaluation of A_I^r resulted in a detected tree length of 51.8% and a false positive rate of 1.01% (indicated as A_I^r in Fig. 6). Fig. 7 shows five 3D examples comparing the proposed method A_I^r to the original EXACT'09 submission A_I.

5.5. Experiment 4

In the final experiment, the aim is to improve the results of a given airway segmentation algorithm by increasing the detected airway tree length while limiting the amount of leaks as proposed in Application 2 (Section 3.2). For this experiment, coarsely-tuned segmentations are extracted with the method described in van Rikxoort et al. (2009) (algorithm 14 of the EXACT'09 challenge). Airway branches are extracted using a wavefront propagation algorithm that includes voxels below a certain threshold t. To prevent the segmentation from leaking, additional rules ensure that the radii ratio between a parent and a daughter branch cannot exceed a fixed value r. Since both t and r have an independent effect on the segmented tree length and the amount of leaks, the interaction between these parameter can produce segmentation with complementary information. We therefore independently varied both t and r in this experiment to extract 15 coarsely-tuned segmentations $\{C_1, \ldots, C_{15}\}$. We applied leak reduction to each of the coarsely-tuned segmentations in order to construct 15 leak reduced segmentations $\{C_1^r, \ldots, C_{15}^r\}$. The union of all C_i^r provided the resulting segmentation A_u^r. The evaluation of A_u^r was performed using the references scans of the EXACT'09 challenge, where all parts of the segmentation that were found by our method, but were not in this references, were visually inspected as was done in the original evaluation of the challenge. This resulted in a detected tree length of 65.4% and a false positive rate of 1.68% (indicated as A_u^r in Fig. 6), which is closest to the optimal combination of all algorithms in the EXACT'09 airway segmentation challenge (indicated by the red star in Fig. 6). Fig. 7 shows five 3D examples comparing A_u^r to A_I.

6. Discussion

Application 1 and 2, as proposed in Section 3, show a practical way of applying our leak reduction method. These applications were evaluated in Experiments 3 and 4 by using the twenty test cases of the EXACT'09 challenge. This test set was substantially different from the set that was used for training the ConvNet, as the test set contained scans from several different scanners, with different scanning protocols and reconstruction parameters. In addition, the test set ranged from clinical dose to ultra low-dose scans, from healthy volunteers to patients with severe lung disease, and from full inspiration to full expiration. The performance in experiment 4 shows that the proposed method can handle these kinds of variation, even though this variation was not available at training time. However, since the training set only consisted of high-dose full inspiratory scans from the multi-center COPDGene study, extending the training set with low-dose scans, expiratory scans, and scans with different reconstruction kernels may further improve the classification and generalization of the method.

In Section 3.2 we propose a strategy to combine coarsely-tuned segmentations, in order to improve the quality of a given airway segmentation algorithm whose performance depends on a set of tunable parameters. Combining multiple segmentations after removing leaks shows to improve the segmentation and in addition

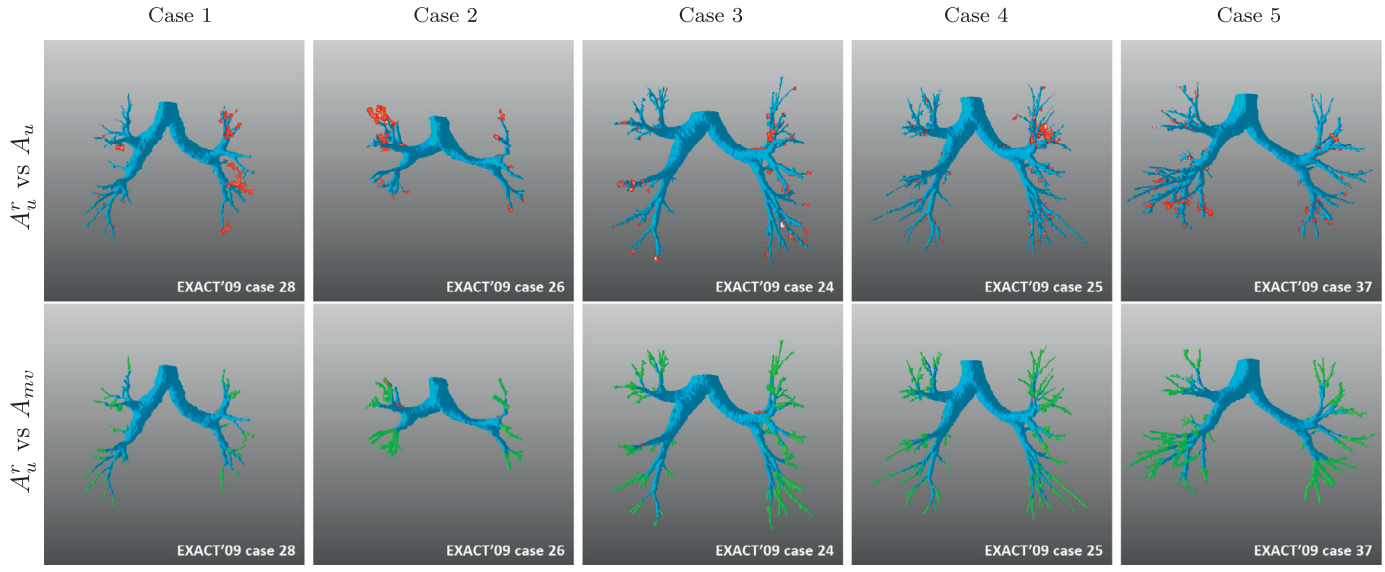

Fig. 8. Five examples taken from Experiment 4 in which the proposed combination strategy of multiple segmentations is compared to two different combination strategies of the same segmentations. In the first row, our combination A_u^r is compared to the union of all coarsely-tuned segmentations A_u, where red indicates voxels that were in A_u, but classified as leak by our proposed method. The blue parts indicate the improved segmentation A_u^r. In the second row our combination A_u^r is compared to majority voting of all coarsely-tuned segmentations A_{mv}, where red indicates voxels that were in A_{mv}, but classified as leak by our proposed method. The green parts indicate voxels that were only included in the improved segmentations A_u^r, whereas blue indicates voxels that were both in A_{mv} and A_u^r. Each rendering was anterior viewed with a field of view of 250x250mm. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

largely circumvents the need for parameter fine-tuning. However, the way of combining airway segmentations is not straightforward and affects the resulting segmentation. In Fig. 8, five cases are shown in which the resulting segmentations from our combination strategy A_u^r are compared to the union and majority voting of all coarsely-tuned segmentations. In the first row of Fig. 8, the difference between A_u^r and the union of the coarsely-tuned segmentations A_u is shown. The second row shows the difference between A_u^r and the majority voting of the coarsely-tuned segmentations A_{mv}. When considering A_u, a substantial amount of leaks is present in the resulting segmentations, whereas A_{mv} produces airway segmentations with low leakage count but also low tree length. The proposed A_u^r on the other hand, shows to reduce the amount of leaks compared to A_u and retains substantially more airways compared to A_{mv}.

In the fourth experiment, we showed that our approach came closest to the optimal combination of the EXACT'09 challenge, however still detected less airways. The main reason for this difference is that only a single algorithm was used to extract the coarsely-tuned segmentations in our approach, whereas the optimal combination of the EXACT'09 challenge consists of segmentations from multiple algorithms. A single segmentation algorithm is able to produce many different segmentations, however may systematically miss a specific part of the airway tree. Since it has been shown in the EXACT'09 study that different segmentation algorithms provide complementary information, a combination of leak reduced segmentations extracted by multiple algorithms may therefore further improve the detection of airways. This suggests that the optimal combination method of EXACT'09 might be further improved by the combination approach proposed in this paper.

The ConvNet that we used for leak detection was specifically designed to classify a small part of a given airway segmentation. Approximating the 3D appearance of a candidate by using the three consecutive patches p_1, p_2, and p_3 was a major design choice for the development of the architecture of the ConvNet. Without the information of these three patches, a leak can easily be misclassified because of an airway-like appearance on a single patch, making the classification task much more challenging. Furthermore, encoding this information into 3 equally spaced 2D patches is an efficient way to represent the airway appearance in the longitudinal view. In addition, parallel processing of the three patches of a candidate allows each separate stack of convolutional layers to retrieve information from the other two patches during training time. This is especially useful in situations where a candidate partially leaks into the parenchyma, in which, for example, two patches are part of the actual airway and one patch is part of a leak. In these kind of scenarios the proposed ConvNet is able to learn to classify this candidate as a leak. Since we do not assume a spatial correspondence between the pixels of each of the three patches, an approach in which we use a single stack of convolutional layers with a three channels input patch would be less suitable. As an alternative to multiple 2D patches, a full 3D approach could be considered instead. However, given the high performance of our 2D approach (i.e. an area under the ROC curve of 0.994) only a small improvement can be expected with a full 3D approach. Nevertheless, exploring the possibilities of 3D ConvNets would be an interesting topic for future work.

Our proposed method may provide the ability to investigate the role of smaller bronchi in pulmonary diseases involving airway pathology. For example, the role of small bronchi in COPD is yet unknown, partially because of a lack of robust airway segmentation algorithms. Especially in these types of scans, the presence of emphysema is the main source of leaks, since emphysema has similar intensity values as the airways. Our proposed method may provide an important tool, because of the ability to limit leakage while increasing the detection of smaller bronchi.

In experiment 3, we showed that our method is able to substantially reduce the amount of leaks in a given airway segmentation. However, this additionally led to a decrease in segmented airway tree length. After careful inspection of these removed airways, we concluded that the majority was because of candidates that represented a combination of both an airway and a leak. In these specific situations the ConvNet may classify the candidate as a leak and consequently remove the leak and the attached airway from the segmentation. A thorough analysis of the false positive candidates suggests that these misclassification especially happens when the severity of leakage is to such an extent that the

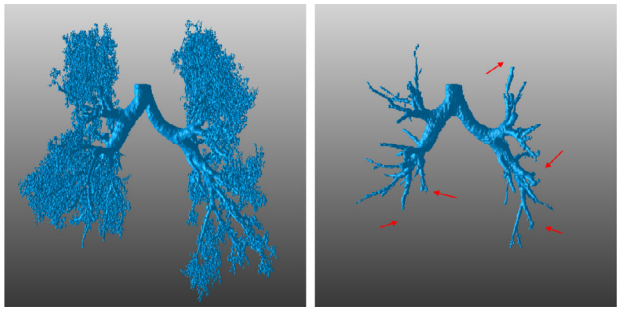

Fig. 9. Example of leaks in a given airway segmentation extracted using a liberal set of parameters (left) and the resulting segmentation after leakage reduction with our proposed method (right). Although the method was profound in removing the leaks from the input segmentation, the red arrows indicate areas where actual airways were falsely classified as leaks and removed from the segmentation. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

appearance of the actual airway branches is affected. An extreme example is shown in Fig. 9, where our method is able to remove the leaks in a given segmentation but in addition removes parts of the actual airway tree. Similar misclassifications may happen in the reverse situation where a candidate that contains both a leak and an airway is classified as airway, resulting in a leak that is not removed from the segmentation. In order to further investigate this, we randomly selected 100 inspiratory CT scans from the COPDGene study with airway segmentation that were visually checked for leaks. After applying Application 1 to these scans, we reduced the average tree length by 1.88%, corresponding to an average of 4.70 mm. Since there were no leaks in the input segmentations, these removed airways are considered to be the actual false positives of our method.

In conclusion, we have presented a method to improve a given airway segmentation by detecting leaks in the segmented airways using ConvNets. Our approach differs from the current state-of-the-art since we do not propose yet another segmentation algorithm, but instead a novel method to detect and remove leaks in a given airway segmentation. Combining multiple segmentations extracted from a single airway segmentation algorithm together with leak reduction has the potential to increase the total tree length of an airway segmentation while limiting the amount of leaks. This way of applying leak reduction achieves a higher sensitivity at a low false-positive rate compared to all the state-of-the-art methods that entered in EXACT09.

Acknowledgements

This work was financed by the Netherlands Organisation for Scientific Research (NWO, project number: 612.001.204). This work was supported in part by NIH grants R01 HL089856 and R01 HL089897. The authors would like to thank the developers of Theano (Al-Rfou et al., 2016; Bergstra et al., 2010).

References

Al-Rfou, R., Alain, G., Almahairi, A., Angermueller, C., Bahdanau, D., Ballas, N., Bastien, F., Bayer, J., Belikov, A., Belopolsky, A., Bengio, Y., Bergeron, A., Bergstra, J., Bisson, V., Bleecher Snyder, J., Bouchard, N., Boulanger-Lewandowski, N., Bouthillier, X., de Brébisson, A., Breuleux, O., Carrier, P.L., Cho, K., Chorowski, J., Christiano, P., Cooijmans, T., Côté, M.A., Côté, M., Courville, A., Dauphin, Y.N., Delalleau, O., Demouth, J., Desjardins, G., Dieleman, S., Dinh, L., Ducoffe, M., Dumoulin, V., Ebrahimi Kahou, S., Erhan, D., Fan, Z., Firat, O., Germain, M., Glorot, X., Goodfellow, I., Graham, M., Gulcehre, C., Hamel, P., Harlouchet, I., Heng, J.P., Hidasi, B., Honari, S., Jain, A., Jean, S., Jia, K., Korobov, M., Kulkarni, V., Lamb, A., Lamblin, P., Larsen, E., Laurent, C., Lee, S., Lefrancois, S., Lemieux, S., Léonard, N., Lin, Z., Livezey, J.A., Lorenz, C., Lowin, J., Ma, Q., Manzagol, P.A., Mastropietro, O., McGibbon, R.T., Memisevic, R., van Merriënboer, B., Michalski, V., Mirza, M., Orlandi, A., Pal, C., Pascanu, R., Pezeshki, M., Raffel, C., Renshaw, D., Rocklin, M., Romero, A., Roth, M., Sadowski, P., Salvatier, J., Savard, F., Schlüter, J., Schulman, J., Schwartz, G., Serban, I.V., Serdyuk, D., Shabanian, S., Simon, E., Spieckermann, S., Subramanyam, S.R., Sygnowski, J., Tanguay, J., van Tulder, G., Turian, J., Urban, S., Vincent, P., Visin, F., de Vries, H., Warde-Farley, D., Webb, D.J., Willson, M., Xu, K., Xue, L., Yao, L., Zhang, S., Zhang, Y., 2016. Theano: a python framework for fast computation of mathematical expressions. (Theano Development Team) arXiv e-prints abs/1605.02688.

Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J., Warde-Farley, D., Bengio, Y., 2010. Theano: A CPU and GPU math expression compiler. In: Proceedings of the Python for Scientific Computing Conference (SciPy).

Breitenreicher, D., Sofka, M., Britzen, S., Zhou, S.K., 2013. Hierarchical discriminative framework for detecting tubular structures in 3D images. In: IPMI, pp. 328–339.

Bülow, T., Wiemker, R., Blaffert, T., Lorenz, C., Renisch, S., 2005. Automatic extraction of the pulmonary artery tree from multi-slice CT data. In: Proceedings of the SPIE, pp. 730–740.

Coates, A., Lee, H., Ng, A.Y., 2011. An analysis of single-layer networks in unsupervised feature learning. In: AISTATS, pp. 215–223.

Fabijańska, A., 2009. Two-pass region growing algorithm for segmenting airway tree from MDCT chest scans.. Comput. Med. Imaging Graph. 33, 537–546.

Fetita, C.I., Prêteux, F., Beigelman-Aubry, C., Grenier, P., 2004. Pulmonary airways: 3D reconstruction from multislice CT and clinical investigation. IEEE Trans. Med. Imaging 23, 1353–1364.

Graham, M.W., Gibbs, J.D., Cornish, D.C., Higgins, W.E., 2010. Robust 3D airway tree segmentation for image-guided peripheral bronchoscopy. IEEE Trans. Med. Imaging 29, 982–997.

Ioffe, S., Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: ICML, pp. 448–456.

Kitasaka, T., Yano, H., Feuerstein, M., Mori, K., 2010. Bronchial region extraction from 3D chest CT image by voxel classification based on local intensity structure. In: Proc. Third International Workshop on Pulmonary Image Analysis, pp. 21–29.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems 25, pp. 1097–1105.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436–444.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proc. IEEE 86, 2278–2324.

Lo, P., van Ginneken, B., Reinhardt, J.M., Tarunashree, Y., de Jong, P.A., Irving, B., Fetita, C., Ortner, M., Pinho, R., Sijbers, J., Feuerstein, M., Fabijanska, A., Bauer, C., Beichel, R., Mendoza, C.S., Wiemker, R., Lee, J., Reeves, A.P., Born, S., Weinheimer, O., van Rikxoort, E.M., Tschirren, J., Mori, K., Odry, B., Naidich, D.P., Hartmann, I.J., Hoffman, E.A., Prokop, M., Pedersen, J.H., de Bruijne, M., 2012. Extraction of airways from CT (EXACT'09). IEEE Trans. Med. Imaging 31, 2093–2107.

Lo, P., Sporring, J., Ashraf, H., Pedersen, J.J.H., de Bruijne, M., 2010. Vessel-guided airway tree segmentation: A voxel classification approach. Med. Image Anal. 14, 527–538.

Palagyi, K., Kuba, A., 1998. A 3D 6-subiteration thinning algorithm for extracting medial lines. Pattern Recognit. Lett. 19, 613–627.

Pu, J., Gu, S., Liu, S., Zhu, S., Wilson, D., Siegfried, J.M., Gur, D., 2012. CT based computerized identification and analysis of human airways: a review. Med. Phys. 39, 2603–2616.

Regan, E.A., Hokanson, J.E., Murphy, J.R., Make, B., Lynch, D.A., Beaty, T.H., Curran-Everett, D., Silverman, E.K., Crapo, J.D., 2010. Genetic epidemiology of COPD (COPDGene) study design. COPD 7, 32–43.

van Rikxoort, E., Baggerman, W., van Ginneken, B., 2009. Automatic segmentation of the airway tree from thoracic CT scans using a multi-threshold approach. In: The Second International Workshop on Pulmonary Image Analysis, pp. 341–349.

van Rikxoort, E.M., van Ginneken, B., 2013. Automated segmentation of pulmonary structures in thoracic computed tomography scans: a review. Phys. Med. Biol. 58, 187–220.

van Rikxoort, E.M., Prokop, M., de Hoop, B., Viergever, M., Pluim, J., van Ginneken, B., 2010. Automatic segmentation of pulmonary lobes robust against incomplete fissures. IEEE Trans. Med. Imaging 29, 1286–1296.

Schmidhuber, J., 2015. Deep learning in neural networks: an overview. Neural Netw. 61, 85–117.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y., 2014. OverFeat: Integrated recognition, localization and detection using convolutional networks. In: International Conference on Learning Representations (ICLR 2014). ArXiv: 1312.6229

Setio, A.A.A., Ciompi, F., Litjens, G., Gerke, P., Jacobs, C., van Riel, S., Wille, M.W., Naqibullah, M., Sanchez, C., van Ginneken, B., 2016. Pulmonary nodule detection in CT images: false positive reduction using multi-view convolutional networks. IEEE Trans. Med. Imaging 35, 1160–1169.

Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv:14091556.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2014. Going deeper with convolutions. arXiv:14094842v1.

Tieleman, T., Hinton, G., 2012. Lecture 6.5-rmsprop: divide the gradient by a running average of its recent magnitude. COURSERA: Neural Netw. Mach. Learn. 4.

Ukil, S., Reinhardt, J.M., 2005. Smoothing lung segmentation surfaces in three--dimensional x-ray CT images using anatomic guidance. Acad. Radiol. 12, 1502–1511.

Zhou, X., Hayashi, T., Hara, T., Fujita, H., Yokoyama, R., Kiryu, T., Hoshi, H., 2006. Automatic segmentation and recognition of anatomical lung structures from high--resolution chest CT images. Comput. Med. Imaging Graph. 30, 299–313.