



# PhD Thesis

Raghavendra Selvan

## Extraction of Airways from Volumetric Data

Advisors: Marleen de Bruijne, Jens Petersen

Handed in: September 30, 2018



## Abstract

Obtaining reliable segmentations of tree structures like airways, vessels and neurons from medical image data can enable important clinical applications. This thesis is concerned with the development of image segmentation methods aimed primarily at obtaining such tree structures from volumetric data. The main focus is on extraction of airways from 3D computed tomography (CT) data.

Most existing airway segmentation methods in literature rely on local and sequential decisions. This renders them susceptible to occlusions and noise in the image data, resulting in missing branches. These concerns are addressed with four overarching themes prevalent in this thesis: exploratory nature of methods, use of relevant global information in making local decisions, use of domain knowledge in segmentation and estimation and incorporation of prediction uncertainty in the decision making process. Adhering to these objectives, our investigations have resulted in four diverse yet related models that achieve these criteria to varying degrees. The proposed tree extraction methods are based on: Multiple Hypothesis Tracking (MHT), Bayesian Smoothing, Mean-Field Networks (MFNs) and Graph Neural Networks (GNNs).

Modifications to an existing interactive vessel segmentation method based on MHT are proposed which turn the original MHT method into an automatic method capable of tracking complete trees starting from a single seed point. The remaining three methods are developed within the framework of probabilistic graphical models. These methods use a sparse graph-like representation of volumetric images based on a two-step preprocessing procedure, involving a trained voxel classifier to obtain airway probability maps followed by multi-scale blob detection. The Bayesian smoothing method uses linear and Gaussian process and measurement models to output candidate airway branches with multivariate Gaussian density estimates as its states. False positive branches from the predictions are discarded by thresholding a measure derived from the uncertainty estimates of the branches. As input for the next two methods, states of these candidate branches are transformed into nodes of a graph with features based on the corresponding multivariate Gaussian densities. Extraction of trees is posed as a graph refinement task, of recovering subgraphs from over-complete graphs such that the subgraphs capture the connectivity between nodes belonging to the underlying airway tree. Two methods based on MFN and GNN are proposed as solutions to the graph refinement task. With MFN, graph refinement is performed as approximate inference using mean-field approximation (MFA). Iterations of MFA are unrolled as feed-forward operations of MFN and gradient descent is used to learn the MFN parameters. Graph refinement using GNN model is performed by jointly training a graph encoder-decoder pair in a supervised learning setup. The encoder learns useful edge embeddings from which the probability of edge connections are predicted using a simple decoder.

Performance of all the methods are evaluated on a subset of CT data from the Danish Lung Cancer Screening Trial, comparing them with manually verified reference segmentations and relevant comparing methods. It is shown that the proposed methods detect more branches with less false positives than baseline methods.

## Resumé

At have adgang til pålidelige segmenteringer af træstrukturer som luftveje, blodkar og neuroner fra medicinsk billeddata kan muliggøre vigtige kliniske anvendelser. Denne afhandling handler om udvikling af billedsegmenteringsmetoder, der primært har til formål at finde sådanne træstrukturer i volumetrisk data. Hovedfokus er på udvindingen af luftveje fra 3D-computertomografidata.

De fleste eksisterende luftvejssegmenteringsmetoder benytter lokale og sekventielle beslutninger. Det gør dem følsomme over for tilstopninger og støj i billeddata, hvilket resulterer i manglende grene. Disse svagheder adresseres med fire overordnede temaer, der er fremherskende i denne afhandling: metodernes sonderende karakter, anvendelse af relevant global information i lokal beslutningstagning, anvendelse af domæneviden i modellerne og estimering og inkorporering af forudsigelsesusikkerhed i beslutningsprocessen. I overholdelsen af disse mål har vores undersøgelser resulteret i fire forskellige men alligevel relaterede modeller, som i varierende grad opfylder disse kriterier. De foreslåede træudvindingsmetoder er baseret på: Multiple Hypothesis Tracking (MHT), Bayesian Smoothing, Mean-Field Networks (MFN) og Graph Neural Networks (GNN).

Modifikationer af en eksisterende interaktiv blodkarssegmenteringsmetode baseret på MHT foreslås; Disse modifikationer gør den oprindelige MHT-metode til en automatisk metode, der er i stand til at spore komplette træer startende fra et enkelt udgangspunkt. De resterende tre metoder udvikles inden for rammerne af probabilistiske grafiske modeller. Disse metoder bruger en sparsom graffignende gengivelse af volumetriske billeder baseret på en foreslået to-trins præprocesseringsprocedure, der involverer en trænet voxel-klassifikator for at opnå sandsynlighedskort efterfulgt af multiskalablobdetektion. Bayesian Smoothing metoden anvender lineære og Gaussiske proces- og målemodeller til at udstede luftvejskandidatgrene med multivariate Gaussiske densitetsestimater som dets tilstande. Falske positive grene fra forudsigelserne kasseres ved at sætte en grænseværdi på et mål for grenenes usikkerhedsestimater. Som input til de næste to metoder transformeres de resulterende kandidatgrene til en graf med beskrivende egenskaber baseret på multivariate Gaussiske densiteter. Træerne findes som en grafforfiningsopgave, hvor delgrafer genoprettes fra overforbundne grafer således, at delgraferne repræsenterer forbindelserne mellem knuderne i det underliggende luftvejstræ. To metoder baseret på MFN og GNN foreslås som løsninger til grafforfiningsopgaven. Med MFN tilnærmes grafforfiningsopgaven ved hjælp af Mean-field Approximation (MFA). Iterationer af MFA rulles ud som MFN feed-forward operationer, og gradient descent bruges til at lære MFN-parametrene. Grafforfining ved hjælp af GNN-modellen udføres ved samtidigt at træne et graf-encoder-decoder-par via superviseret læring. Encoder lærer nyttige indlejringer, hvorfra sandsynligheden for kantforbindelser forudsiges ved hjælp af en simpel dekoder.

Ydelsen af alle metoderne evalueres på en delmængde af data fra Det Danske Lunge Screenings Studie, i sammenligning med manuelt verificerede referencesegmenteringer og relevante baseline-metoder. Det er vist, at de foreslåede metoder kan finde flere grene med færre falske positive resultater end sammenligningsmetoderne.

## Acknowledgements

Sticking to the cliché, I will also have to admit that the three years of the PhD project have gone by in a flash. Feels like I stepped into Sigurdsgade 41 for the first time only a couple of months ago. It has turned out to be an extraordinary journey starting from when I first flew out of my cocooned home in Bangalore way back in 2013. And immense gratitude is due to a lot of people who have made this milestone possible for me.

Foremost are my advisors, Marleen and Jens, for believing in me and offering me this amazing opportunity to be working alongside them. Marleen for being such an inspiration with her acumen, diligence and thoughtfulness, and Jens for his meticulousness, patience and all the stimulating discussions. This project would not have been possible without their incessant support at each and every step by encouraging my ideas while at the same time critically evaluating them. It has been an exhilarating learning experience on a daily basis working with them and I can only hope I will be able to emulate them to some extent in the future.

Special thanks to Max Welling for hosting me at University of Amsterdam during my research visit and giving a major boost to the project. I would also like to acknowledge and appreciate the efforts of Thomas Kipf for our collaboration.

I would also like to thank our funding agencies Independent Council for Research (DFF) and Netherlands Organisation for Scientific Research (NWO).

Working on the PhD project has been intense and while I was diving deep into it, several factors acted as counterweights so that I did not get lost into the abyss. My vibrant and actual social network in Copenhagen was a major part of this. Special shout-out to Liz, Kristoffer, Line, Niels, Niklas, Henrik, Jacob (for Kungsleden and Slipknot!), Silas, Mareike, Yova, Ana, Anton, Irme (sorted based on seniority of knowing me :P) and everyone else at the former Image Section. Not just for the fun aspects but for being around as considerate friends who actually care. And, for fun too! During the course of my PhD I ended up developing two new deep passions that have helped me stay calm and composed. First is running; so cheers to all my running mates. Second is Pyotr Illich Tchaikovsky; love to the maestro for speaking to me in volumes with his music.

Finally, thanks to all the women in my life who have, and are, molding me. Especially to my mother for being such a hero, my sister for being a great pillar of strength and my partner, Sneha, for being the most understanding and loving person, ever.

Here's a timeless quote from one female Tamil poet Avvaiyar (circa 3rd Century BCE), to philosophically contextualise this thesis. It roughly translates to

*“Known is a drop, unknown is an ocean.”*

கற்றது கைமண் அளவு, கல்லாதது உலகளவு



# Contents

<b>Abstract</b>	<b>i</b>
<b>Acknowledgements</b>	<b>iii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Anatomy of human pulmonary system . . . . .	1
1.2 Chronic obstructive pulmonary disease . . . . .	3
1.3 Study of airways using imaging and CAD . . . . .	4
1.4 Overview of airway segmentation methods . . . . .	5
1.5 Outline of thesis . . . . .	7
1.6 Main contributions . . . . .	9
<b>2 Multiple Hypothesis Tracking</b>	<b>11</b>
2.1 Introduction . . . . .	12
2.2 Method . . . . .	15
2.3 Experiments and Results . . . . .	19
2.4 Discussion and Conclusions . . . . .	28
<b>3 Bayesian Smoothing</b>	<b>31</b>
3.1 Introduction . . . . .	32
3.2 Method . . . . .	33
3.3 Experiments and Results . . . . .	39
3.4 Discussion and Conclusions . . . . .	42
<b>4 Graph Refinement Methods</b>	<b>43</b>
4.1 Introduction . . . . .	44
4.2 Methods . . . . .	46
4.3 Experiments and Results . . . . .	52
4.4 Discussion and Conclusions . . . . .	60

<b>5 Summary and Discussion</b>	<b>67</b>
5.1 Summary . . . . .	67
5.2 Discussion and conclusion . . . . .	69
5.3 Future prospects . . . . .	73
<b>List of Publications</b>	<b>75</b>
<b>Bibliography</b>	<b>77</b>



# Chapter 1

## Introduction

Tree structures belonging to vasculature, neuronal tracks and airways are constantly analysed in medical images requiring substantial effort from trained professionals like radiologists. Automating bulk of such analysis can relieve radiologists of their work, while also providing additional insights, to make better diagnostic decisions. Recent advances in image analysis methods have opened up possibilities of augmenting human experts in this regard. Aligned with this objective, this thesis concerns itself with the extraction of tree structures from volumetric data and, in particular, is focused on studying airway trees.

### 1.1 Anatomy of human pulmonary system

The primary function of the human pulmonary system is to enable gas exchange. The respiratory system comprises nose, nasal passages, pharynx, portion of larynx in the upper respiratory tract. The lower respiratory tract comprises the airway tree starting from the trachea, lungs, pulmonary vessels, the lung tissue and diaphragm. The right lung has three lobes: superior, middle and inferior, divided so by two fissures. In comparison, the left lung has only two lobes: superior and inferior. These lobes form the five basic regions of the lungs. These lobes are further divided into bronchopulmonary segments. The right lung has 10 bronchopulmonary segments: three in the superior lobe, two in the middle lobe and five in the inferior lobe. The left lung has eight bronchopulmonary segments: four in each of its lobes. The lung tissue, also known as the lung parenchyma, consists of up to 700 million specialised units called alveoli, which are responsible for gas exchange [Roberts et al., 2000]. An anatomical view of the human pulmonary system is shown in Figure 1.1.

The airway tree is, for the most part of it, a fractal like branching network of air conducting tubular passage [Weibel, 2009]. It begins at the oral and

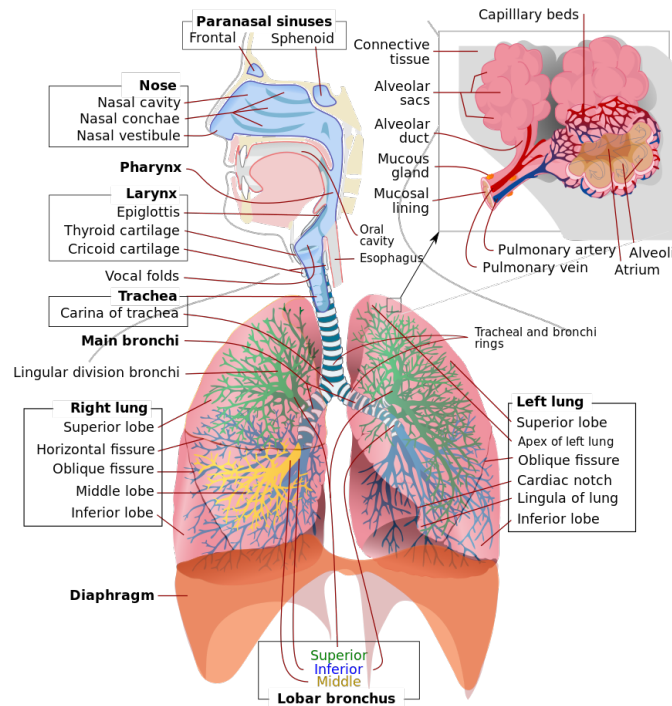


Figure 1.1: Anatomical view of the human pulmonary system. (Image sourced from [https://commons.wikimedia.org/wiki/File:Respiratory\\_system\\_complete\\_en.svg](https://commons.wikimedia.org/wiki/File:Respiratory_system_complete_en.svg))

nasal cavities, and leads into the trachea which bifurcates as the two primary bronchi entering the lungs. The primary bronchi then continue branching, into secondary bronchi, tertiary bronchi and then into smaller airways called bronchioles which branch further, until they terminate into the alveoli. One distinguishing factor between bronchi and bronchioles, apart from their sizes, is the presence and absence of cartilage, respectively. An adult human respiratory system can have around 23 generations of airways [Weibel and Gomez, 1962]. A typical branching of a human airway tree showing few generations of airways is shown in Figure 1.2.

Respiratory diseases such as asthma [Lipworth et al., 2014], cystic fibrosis [Wielpütz et al., 2013] and COPD [Hogg et al., 2017] adversely affect airways, hampering their functionality and also the general well-being of the affected subjects.

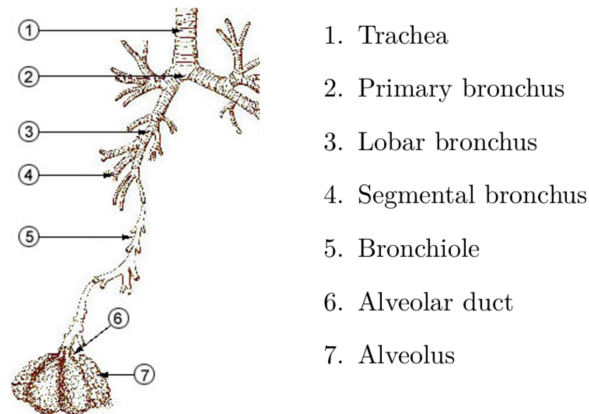


Figure 1.2: Branching in a typical human airway tree. Adapted from [https://commons.wikimedia.org/wiki/File:Illu\\_quiz\\_lung05.jpg](https://commons.wikimedia.org/wiki/File:Illu_quiz_lung05.jpg)

## 1.2 Chronic obstructive pulmonary disease

Chronic obstructive pulmonary disease (COPD) is a leading cause of morbidity and mortality worldwide. Currently COPD is the fourth leading cause of death worldwide and is projected to be the third leading cause, as early as 2020 [GOLD, 2017]. It is estimated that about 200 million people are affected by COPD and has claimed an estimated 3.5 million lives in 2015 [Vos et al., 2016]. Tobacco smoking, indoor and outdoor air pollution, chemicals and occupational dusts are known to be the main causes of COPD [Cruz, 2007].

COPD is an umbrella term used to characterise the destruction of lung tissue (emphysema) and irreversible changes to airway morphology. The airway tree is adversely affected by COPD in several ways: inflammation and narrowing of peripheral airways, thickening of airway walls, collapsing of airways, mucus plugs in smaller airways and loss of small airways [GOLD, 2017]. These pathophysiological abnormalities limit airflow and gas exchange capabilities of the pulmonary system.

The primary method to diagnose COPD is based on pulmonary function tests (PFTs) such as spirometry. Spirometry measures the volume and/or flow of air during inspiration and expiration. It is the most widely used and objectively reproducible measurement of airflow limitation [GOLD, 2017]. Two critical measures obtained from spirometry are used to diagnose COPD. Forced Expiratory Volume in one second ( $FEV_1$ ) and Forced Vital Capacity (FVC), measure the volume of air a subject can forcibly exhale after full inspiration in the first second and in entirety, respectively. The ratio of these two measures,

Table 1.1: Classification of airflow limitation severity in COPD, in patients with  $FEV_1/FVC < 0.7$ , according to GOLD 2017

Class	Severity	Lung function
GOLD 1:	Mild	$FEV_1 > 80\%$ predicted
GOLD 2:	Moderate	$50\% \leq FEV_1 < 80\%$ predicted
GOLD 3:	Severe	$30\% \leq FEV_1 < 50\%$ predicted
GOLD 4:	Very Severe	$FEV_1 < 30\%$ predicted

$FEV_1/FVC$ , is a useful indicator of lung function. According to the Global Initiative for Chronic Obstructive Lung Disease,  $FEV_1/FVC < 0.7$  indicates airflow limitation and the subjects are recommended to undergo COPD diagnosis. These subjects are then categorised into four COPD severity classes based on the predicted  $FEV_1$  as shown in Table 1.1 [GOLD, 2017].

Spirometry tests are non-invasive, inexpensive and mostly reliable in the diagnosis of COPD. However, lung function tests do not reveal the underlying pathophysiological changes that have occurred. In some cases, when  $FEV_1$  and FVC have proportionally reduced the  $FEV_1/FVC$  ratio does not decrease and early diagnosis of COPD cannot be achieved. Preemptive imaging based qualitative analysis of airways, in case of regular smokers for instance, can overcome these constraints providing a more holistic setting for the diagnosis and prognosis of COPD [Ley-Zaporozhan et al., 2008, Hogg et al., 2013]. Also, imaging techniques allow localising and subtyping of disease.

### 1.3 Study of airways using imaging and CAD

Computer aided diagnosis (CAD) has become indispensable within 50 years of its inception and subsequent integration at the radiologist's desk [van Ginneken, 2017]. Today, further advances in high-resolution imaging and newer classes of algorithms have ushered in enormous possibilities for CAD. Imaging based diagnosis using chest computed tomography (CT) has been applied to perform diagnosis and prognosis of several cardio-vascular and pulmonary diseases, including COPD. Figure 1.4 shows a low-dose, volumetric chest CT scan along with views in the axial, coronal and sagittal planes.

Morphometric analysis of airway trees using imaging techniques is being studied extensively to derive COPD biomarkers for early and reliable diagnosis [McDonough et al., 2011, GOLD, 2017]. Following the discussion in [San Jose Estepar et al., 2008], one possible view of morphometric analysis of airway trees is to see it as a pipeline comprising three distinct steps: detection, localisa-

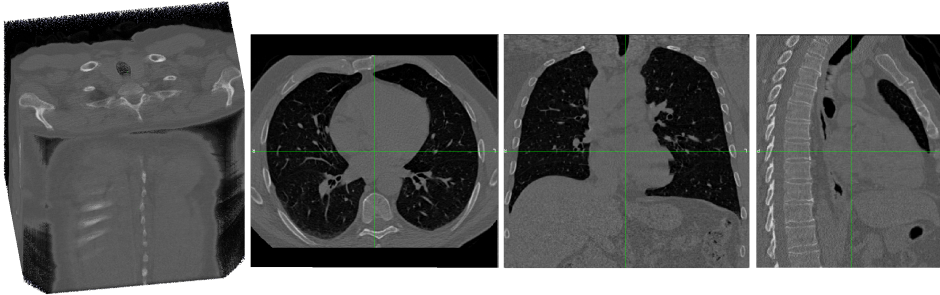


Figure 1.3: Volumetric and three-axial views of a low-dose chest scan that can be used to study lungs for diagnosing and staging of COPD.

tion and validation. The foremost challenge is to detect airways from the image data. This is all the more crucial in relation to COPD as the significant damage is caused to small airways. Once the airways are detected then computing local measures like the airway wall thickness, lumen diameter and other interesting morphological properties is seen as localisation. Using these properties to validate the prevalence of disease either by correlating with lung function tests or in some cases with histopathological information is the validation step.

This thesis primarily concerns itself with the first of these steps: detection, especially, of small and/or occluded airways by developing novel image segmentation methods that are robust to these challenges.

## 1.4 Overview of airway segmentation methods

Image segmentation is a fundamental step in image analysis. It is the task of partitioning images into meaningful regions. One can also view image segmentation from a signal processing point of view, and observe that the regions of interest are, in fact, the underlying signals to be filtered out from noise. For instance, in the chest CT data shown in Figure 1.4, depending on the application one might be interested in segmenting the heart, entire lung regions, airways, vessels, or even the spine or ribs. In simple segmentation settings, when the task is to segment one of the organs all remaining information in the data can be interpreted as the first source of noise. Second category of noise, can be attributed to the image acquisition process. In case of low-dose CT images, the signal-to-noise ratio can be poor and consequently it becomes harder to detect and segment smaller structures. This is further aggravated by imaging artifacts, induced by motion and variations across different scanners. A third source of noise could be anatomical and pathological variations across subjects.

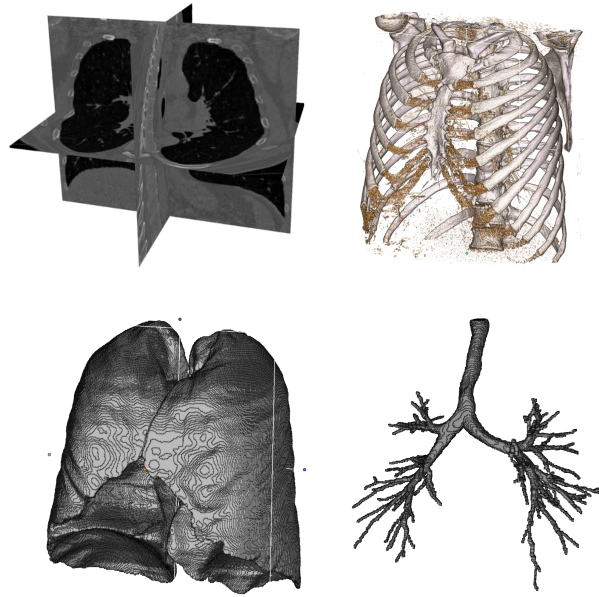


Figure 1.4: A chest CT scan visualised in its 3-axes (top left) along with segmentations of bones (top right), lungs (bottom left) and the airway tree (bottom right), obtained from the scan. The visualisations and bone segmentation were generated using 3DSlicer, <https://www.slicer.org/>

As an example, when the target regions are airways then mucus plugs within the airways due to disease can make the segmentation challenging. Thus, the seemingly simple task of image segmentation poses several interesting challenges. These challenges have been addressed in literature for various applications with a diverse set of methods [Pal and Pal, 1993, Pham et al., 2000, Setarehdan and Singh, 2012].

A concise overview of segmentation methods that focus on chest CT data to segment different structures is provided in [Van Rikxoort and Van Ginneken, 2013, van Ginneken, 2017], and a more focused survey of airway segmentation methods in [Pu et al., 2012, Lo et al., 2012]. We next summarise the broad classes of airway lumen segmentation methods and bridge them with the more recent methods.

Region-growing is the most basic of segmentation methods that operates by iteratively adding voxels that are similar to the initial seed points to the segmented region until no more voxels meet the similarity criterion. The simplest criterion, which has been quite successful in airway segmentation, is to define regions that are similar in their intensity values to those of the seed points.

Other specialised similarity criteria can also be used for growing regions, which has been a primary theme with the class of region-growing based methods used for airway segmentation. In the EXACT challenge [Lo et al., 2012], out of the 15 airway segmentation methods that were compared ten of them use some form of region-growing. One of these methods in [Lo et al., 2010], recurs throughout this thesis as a comparing method and we make a mention of it here. This method performs region growing on probability images obtained using a k-NN based voxel classifier using multi-scale Gaussian derivative features, trained to distinguish airway and non-airway voxels. Further, based on an additional criterion airway candidates with lower probability that are sufficiently similar in orientation to that of a nearby vessel are added to the airway segmentation.

Prior knowledge about the morphology of airways like their branching angle, shape and size at different generations can be used to improve segmentations [Kitaoka et al., 1999]. This idea is exploited in template based methods wherein, simple templates such as tubular ones are tested with the image data to detect likely airway regions. These template matchings are then linked using an iterative or sequential tracking approach to recover the airway trees [Bauer et al., 2009, Born et al., 2009]. In [Bauer et al., 2015], the candidate airway regions are obtained using a tube detection filter and the airway tree is reconstructed using a graph-based optimisation. More recently, in [Zhao et al., 2017], tried to incorporate tree statistics as a prior in addition to using airway appearance model based priors within their template based framework.

Recently, with the advancement of machine learning techniques newer methods have been proposed to segment airways. A random forest based voxel classifier along with an automatic sampling procedure optimised to generate samples based on incomplete reference standard was proposed in [Bian et al., 2018]. The first successful demonstration of using convolutional neural networks in airway segmentation was to improve airway segmentation, from coarsely tuned binary segmentation methods like intensity region growing, by removing leakages in [Charbonnier et al., 2017]. Since then, a 3D fully convolutional network has been used to obtain coarse segmentations from incomplete labels with a graph based refinement step in [?]. A 3D U-net based method used along with iterative tracking was presented in [Meng et al., 2017] and a recent work uses 3D U-net to predict airways directly based on end-to-end training [Juarez et al., 2018].

## 1.5 Outline of thesis

When the work on this project commenced in 2015, the most successful airway segmentation methods in literature were still primarily based on making local

decisions, like the region-growing and template-matching based methods reported in [Lo et al., 2012]. An early, and later on persistent, motivation for the work presented in this thesis was to develop methods that would solve the airway tree extraction problem from a global perspective, by incorporating relevant global information when making local decisions. Further, use of Bayesian modelling to use prior domain knowledge, and estimation and incorporation of uncertainty in the decision making process was the secondary motivation. Throughout this project, the investigations performed have been consistent with these themes, resulting in four novel methods that can be used to segment tree structures from volumetric image data.

In Chapter 2, we present our first method based on multiple hypothesis tracking that makes sequential segmentation decisions by deferring them to a future step, by maintaining a hypothesis tree. Deferred decisions take more global information into account when making local decisions. We adapt an existing interactive method, that has been successfully used to segment coronary arteries and propose modifications to it, making it fully automatic. We demonstrate its usefulness to extract tree structures including airways and coronary arteries in automatic settings.

The remaining three methods, presented in Chapters 3 and 4, are closely related and fall under the category of methods that are, broadly, based on probabilistic graphical models.

In Chapter 3, we take up a Bayesian tracking approach to extracting candidate airway branches. We propose to transform the volume image into a sparse graph-like representation based on simple multi-scale blob detection. We perform Bayesian smoothing, using linear Gaussian process and measurement models, to obtain candidate airway branches. These candidate airway branches are validated based on the inherent uncertainty estimates provided by the Bayesian smoothing procedure.

In Chapter 4, we pose the task of validating candidate branches obtained from the Bayesian smoothing method as a graph refinement task. That is, we assume that the structure of interest is a subgraph embedded in an over-complete input graph. We propose two methods to solve the graph refinement task in this chapter.

In the first graph refinement method, presented in Chapter 4, we pose the extraction of the desired subgraph as an approximate Bayesian inference task. We model the node and pairwise behaviour of nodes based on our prior knowledge as parameterised potential functions and perform inference using iterations of mean-field approximation (MFA). We show that by unrolling the MFA iterations as feedforward operations of a neural network, the parameters of this model can be learnt by performing gradient descent, and call it the



mean-field network (MFN).

In the second graph refinement method, presented in Chapter 4, we take up a supervised learning approach using graph neural networks (GNNs). We present a graph auto-encoder that uses an encoder that can process graph-structured data as input and learns useful edge embeddings from training data. Then, a simple decoder is used to predict edge probabilities from the learnt edge embeddings.

## 1.6 Main contributions

The main contributions of this thesis are listed below:

- Proposed modifications to an existing, interactive, vessel tracking method to make it automatic and a novel application of this method to extract airway trees. Introduced statistical ranking of hypotheses used in the multiple hypothesis tracking method, yielding a probabilistic interpretation of the relative quality of the local hypotheses. Performed comprehensive evaluation of this modified method to extract airways and coronary arteries.
- Novel application of Bayesian smoothing to extract candidate airway branches from volume data, using proposed linear and Gaussian process and measurement models. Experiments on airway extraction show that the proposed exploratory method is able to detect more branches than the comparing methods and the estimated total variance of extracted branches can be used to remove false positives (Chapter 3).
- Proposed a pipeline to transform volume data into graph-like format with nodes associated with multivariate Gaussian densities using Bayesian smoothing. This yields a sparse yet informative representation of dense volumetric images. Based on these graph inputs and initial connectivity between nearest neighbours, extraction of tree structures is posed as a graph-refinement task (Chapters 3 and 4).
- Approximate Bayesian inference based graph refinement procedure using mean-field approximation (MFA) was proposed. MFA assumes a factorisable class of distributions that approximate the posterior distribution over all possible subgraphs and yields an iterative message passing procedure for inference. Proposed node and pairwise potentials that model our prior knowledge about nodes in airway trees. Learning of parameters of this model was done using gradient descent on a feedforward network, as a mean-field network obtained by unrolling the MFA iterations. Performed

centerline distance based comparison with other relevant methods and highlight the advantages of MFN (Chapter 4).

- Proposed a Graph Auto-Encoder based solution to perform graph refinement that uses an encoder to learn useful edge embeddings. A simple decoder is used to predict the probability of connections between nodes from the learnt edge embeddings. Experiments to extract airways show that the GNN model is able to perform significantly better than the comparing methods based on centerline distance measure (Chapter 4).
- Presented connections between MFN and GNN models. We argue that GNNs can be seen as generalisation of message passing algorithms like the MFN and, conversely, the MFN can be seen as a structured neural network based on the underlying graphical model that is built on our prior knowledge. Depending on the availability of prior knowledge (MFN) and sufficient training data (GNN) one of the two models can be more suited for different applications (Chapter 4).

## Chapter 2

# Multiple Hypothesis Tracking

The work presented in this chapter is based on paper that has been submitted as [1]. Portions of the work in this chapter are based on [2]:

- [1] Raghavendra Selvan, Jens Petersen, Jesper H. Pedersen, and Marleen de Bruijne. “Extracting Tree-structures in CT data by Tracking Multiple Statistically Ranked Hypotheses”. (2018)
- [2] Raghavendra Selvan, Jens Petersen, Jesper H. Pedersen, and Marleen de Bruijne. “Extraction of airway trees using multiple hypothesis tracking and template matching”. In *The Sixth International Workshop on Pulmonary Image Analysis. MICCAI, 2016.*

## Abstract

In this work, we adapt a method based on multiple hypothesis tracking (MHT), that has been shown to give state-of-the-art vessel segmentation results in interactive settings, for the purpose of extracting trees in an automatic manner. Regularly spaced tubular templates are fit to image data forming local hypotheses. These local hypotheses are used to construct the MHT tree, which is then traversed to make segmentation decisions. Some critical parameters in the method, we base ours on, are scale-dependent and have an adverse effect when tracking structures of varying dimensions. We propose to use statistical ranking of local hypotheses in constructing the MHT tree which yields a probabilistic interpretation of scores across scales and helps alleviate the scale-dependence of MHT parameters. This enables our method to track trees automatically starting from a single seed point. Our method is evaluated on chest CT data to extract airway trees and coronary arteries. In both cases, we show that our method performs significantly better than the original MHT method.

*Index terms* – multiple hypothesis tracking, tree segmentation, CT, airways, vessels

## 2.1 Introduction

Extracting tree structures is a commonly encountered task in image analysis applications. In medical image analysis, reliable methods to extract airways, blood vessels and neuron tracks have important clinical usage. For instance, airway tree extraction is used to study the morphology of airways, which is useful to derive biomarkers for diseases such as chronic obstructive pulmonary disease (COPD) [Nakano et al., 2000, Hasegawa et al., 2006] and cystic fibrosis [Kuo et al., 2017]; and segmentation of coronary vessels is useful in prognosis of cardio-vascular diseases [Rosamond et al., 2008].

Several methods have been proposed to address tree segmentation tasks occurring in medical images. One general approach is to model local appearance of structures in the tree and the overall connectivity of these local structures. A common local modelling approach is to use templates of the ideal structures of interest and perform template-matching with the image data. This accentuates possible local signals of interest and additional processing on top of such template-matched data can be used to extract the overall connectivity. Multi-scale vesselness filtering is a commonly employed technique to enhance local tubular structures [Frangi et al., 1998, Bauer and Bischof, 2008, Yang et al., 2012] like vessels and airways. Another simpler approach to detect local structures is to use multi-scale blob detectors [Selvan et al., 2017].

Template-matching enhances local structures of interest but invariably ends up being noisy. Several classes of methods are used to extract global structures by processing such noisy data. Minimal path methods can be used to extract tree structures by first assigning costs based on template-matching and computing a path of minimum cost via some optimisation scheme. In Wink et al. [Wink et al., 2004], multi-scale vesselness filter output is transformed into path costs and minimum cost paths are computed using a standard algorithm based on Dijkstra [Dijkstra, 1959], yielding candidate vessel centerlines. Another related minimal path based formulation is presented in [Benmansour and Cohen, 2011], where a customised energy function is minimised that takes the vessel orientation and tubular structure into account. This method is interactive as it obtains minimal paths between user placed points. Another commonly used approach is to employ tracking on top of template-matching. From a signal processing point of view, tracking methods can be seen as powerful denoising techniques that can incorporate strong models. In [Selvan et al., 2017], Bayesian smoothing is performed using a Kalman smoother to denoise super-voxels from a blob-detector to obtain candidate airway branches. Particle filters have also been used in vessel segmentation applications with template-matching performed at each prediction step [Florin et al., 2005, Lesage et al., 2016]. Particle filters can be interpreted as methods that sample multiple hypotheses in a probabilistic setting to make improved segmentation decisions. They are found to be computationally expensive in most applications due to the sampling and testing of numerous hypotheses [Lesage et al., 2009]. Recent methods involving deep learning, such as in [Meng et al., 2017] used to extract airways, also take up a tracking-like approach with neural networks used as learnable template functions.

A comprehensive survey of vessel segmentation methods is presented in [Lesage et al., 2009]. The class of methods we focus on in this contribution are what the authors in [Lesage et al., 2009] address as template-matching based centerline tracking methods. A comparative study for airway segmentation methods was published in the EXACT'09 study [Lo et al., 2012], wherein several methods were evaluated on chest CT data. An important take-away from this study has been that there was scope for improvement in extracting missing airway branches. Another interesting point from the study was that more than half of the competing methods used some form of region growing in their segmentation procedure [Feuerstein et al., 2009, Bauer et al., 2009, Wiemker et al., 2009, Lee and Reeves, 2009].

In this work, we investigate the feasibility of multiple hypothesis tracking (MHT) [Reid, 1979, Blackman, 2004] – a well known decision making paradigm in multi-target tracking – for segmenting trees. In methods based on region growing [Lo et al., 2010] and conventional object tracking [Yedidya and Hartley,

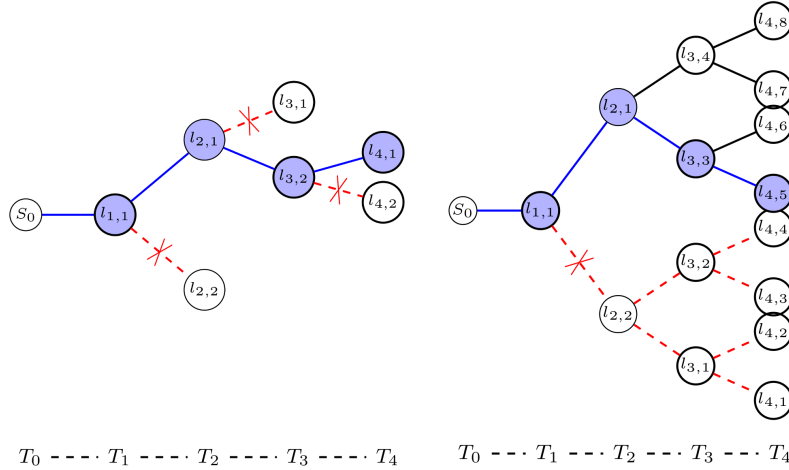


Figure 2.1: Instantaneous and deferred decisions illustrated using hypothesis trees. Hypothesis trees are shown for tracking steps  $T_0, \dots, T_3$  with local hypothesis  $i$  at step  $t$  is denoted by  $l_{t,i}$ . In case of instantaneous decisions (left) only the best local hypothesis at each instant is retained (marked in blue). In the case of MHT with search depth 3 (right), the decision at  $T_1$  is based on the best global hypothesis at step  $T_3$ , marked with blue edges. Once the decision at  $T_1$  has been made, hypotheses that are not children of the best node are discarded, shown in red.

2008], segmentation decisions are made instantaneously making them susceptible to local anomalies due to acquisition noise, interfering structures and others. The primary contrasting feature of using MHT is that it is based on deferring decisions to a future step. From any given step, all possible states corresponding to different solutions (local hypotheses) are maintained up to a predefined search depth (global hypotheses) in the form of a hypothesis tree. Decisions about which of the hypotheses to propagate and which ones to prune away at each step are made by tracing back the best hypothesis from the end of the hypothesis tree. Such deferred decisions, invariably, take more information into account and make the decisions at any given step more robust than instantaneous ones. The concept of instantaneous and deferred decisions are illustrated in Figure 2.1.

In the work of [Friman et al., 2010], MHT was used along with template matching for segmenting small and bright tubular structures from a dark background. This method used in an interactive setting along with minimal paths has remained the best performing method in a coronary vessel segmentation challenge [Schaap et al., 2009], but has not been applied to other tree extraction

problems. Critical parameters used to maintain the MHT tree in this method are scale dependent; this has an adverse effect when tracking structures of varying dimensions, such as airway trees. We believe this method has potential to solve a wider range of problems if the limitations with scale dependence can be alleviated.

We introduce statistical ranking of local hypotheses as a means to make local and global hypothesis scores independent of scales. This makes tuning of two crucial MHT parameters easier, allowing us to apply the method to extract trees and/or structures with branches of varying dimensions without user interaction. The work presented here is based on [Friman et al., 2010] and an extension of our previously published work [Selvan et al., 2016]. Compared to [Selvan et al., 2016], we present the proposed modifications formally and perform more comprehensive evaluation. We show airway extraction experiments on low-dose CT data from the Danish lung cancer screening trial [Pedersen et al., 2009] and segmentation of coronary arteries from CT angiography data from the coronary challenge [Schaap et al., 2009].

## 2.2 Method

In this section, we present an overview of the method in. [Friman et al., 2010], describe its limitations and present modifications that enable our proposed method to overcome these limitations. For convenience, the method in [Friman et al., 2010] will be addressed as the original MHT method and ours, with the modifications, as the modified MHT method.

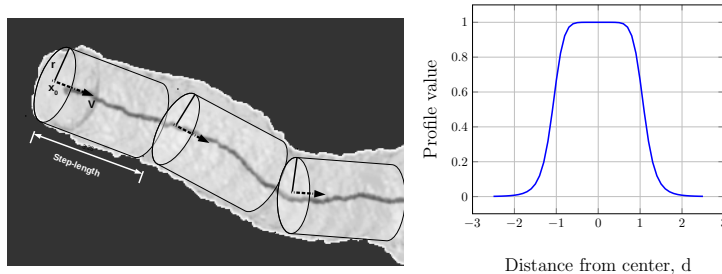


Figure 2.2: Illustrations of approximating an elongated structure of interest with a sequence of tubular segments (left) and a sample profile function used to construct tubular templates (right).

### 2.2.1 The original MHT method [Friman et al., 2010]

The primary objective formulated in the original MHT method is to track bright tubular structures in darker background based on the relative contrast of the structures of interest. Elongated structures are modeled as sequences of tubular segments, as depicted in Figure 2.2, and are segmented by analysing a multiple hypothesis tree comprising template-matched local hypotheses. From each tracking step, several local hypotheses are tested with image data at regular step lengths controlled by the *step length factor* (used as a scaling of the local radius to obtain actual step length). The search span and number of local hypotheses at each step are controlled by the *search angle* and *number of angles* parameters, respectively.

A template function is used to map cross-sectional intensity variations of a tubular template, of radius  $r$  centered at  $\mathbf{x}_0$  along the direction  $\hat{\mathbf{v}}$ , to a profile function with values between 0 and 1, given as,

$$T(\mathbf{x}; \mathbf{x}_0, \hat{\mathbf{v}}, r) = \frac{r^\gamma}{(d_E^2(\mathbf{x}; \mathbf{x}_0, \hat{\mathbf{v}}))^{\gamma/2} + r^\gamma}. \quad (2.1)$$

Steepness of the profile function is controlled by  $\gamma$  and  $d_E^2(\mathbf{x}; \mathbf{x}_0, \hat{\mathbf{v}})$  is the squared Euclidean distance between any point  $\mathbf{x} \in \mathbb{R}^3$  and axis of the tubular template along  $\hat{\mathbf{v}}$ . We retain  $\gamma = 8$  from Friman et al. [Friman et al., 2010] and the corresponding profile function is shown in Figure 2.2 (right).

The image neighbourhood model,  $\mathbf{I}(\mathbf{x})$ , used to fit the template in Eq. (2.1) with image data is given as,

$$\mathbf{I}(\mathbf{x}) = kT(\mathbf{x}; \mathbf{x}_0, \hat{\mathbf{v}}, r) + m + \epsilon(\mathbf{x}), \quad (2.2)$$

where  $\epsilon(\mathbf{x})$  captures noise which could be due to interfering structures, image artifacts or acquisition noise,  $k$  is the contrast and  $m$  is the mean intensity. Template fitting is carried out by solving the following weighted least-squares problem:

$$\arg \min_{k, m, r, \mathbf{x}_0, \hat{\mathbf{v}}} \|\mathbf{W}(r, \mathbf{x}_0, \hat{\mathbf{v}})[kT(\mathbf{x}_0, \hat{\mathbf{v}}, r) + m\mathbf{1}_n - \mathbf{I}]\|^2, \quad (2.3)$$

where  $\mathbf{W}(r, \mathbf{x}_0, \hat{\mathbf{v}})$  is a weighting function with diagonal entries corresponding to an asymmetric Gaussian centered at  $\mathbf{x}_0$ . It is used to localise the fitting procedure and  $\mathbf{I}$  is the image data with non-zero weights under this weighting function. The width of the weighting function is controlled by the *weight window* parameter. We refer to the original paper [Friman et al., 2010] for the procedure to solve this minimisation problem, which is carried out using the Levenberg-Marquardt algorithm.



The template fitting procedure in (2.3) yields updated set of tubular template parameters –  $r, \mathbf{x}_0, \hat{\mathbf{v}}$  – and an estimate of the mean intensity ( $m$ ) and contrast ( $k$ ) of the updated template in the image. Updated tubular templates form the local hypotheses used in the MHT tree. The fitness of local hypotheses with image data is measured using the updated parameters by computing the local hypothesis score:

$$l_i \triangleq \frac{k - m}{\text{std}(k)}. \quad (2.4)$$

The local hypothesis score is a measure of how significantly different the bright tubular structure is from its background; this can be interpreted as the contrast signal-to-noise ratio (SNR), used to compare different local hypotheses. Local hypotheses below a threshold, referred to as the *local threshold*, are discarded to control the number of candidate hypotheses. An MHT tree of a predetermined search depth,  $d$ , is constructed using local hypotheses from each step. A sequence of local hypothesis of length  $d$  forms a global hypothesis with an average global score,

$$s_g = \frac{\sum_{i=1}^d l_i}{d}. \quad (2.5)$$

This score is compared with the *global threshold* and all global hypotheses that do not exceed this threshold are discarded.

In summary, two categories of parameters are used to tune the original MHT method. First category of parameters are related to generating local hypotheses: minimum and maximum radii, step length factor between successive tracking steps, window of the weighting function, maximum search angle and number of local hypotheses at each step. Second category of parameters are used to control the MHT behaviour: search depth, local and global thresholds.

### 2.2.2 Modifications to the original MHT method

The original MHT method was devised as an interactive method to track elongated structures. It is not immediately applicable for the automatic extraction of trees with branches of varying dimensions. In this section, some limitations of the original MHT method in this regard are elaborated and modifications are proposed to overcome them.

#### Dealing with scale-dependence

An important factor to consider when tracking tree-structures with branches of varying dimensions is their range. With airway trees for instance, the radius of visible airways in CT typically ranges from 1mm to 10mm. When using the original MHT method, such variations make it hard to obtain optimal parameter

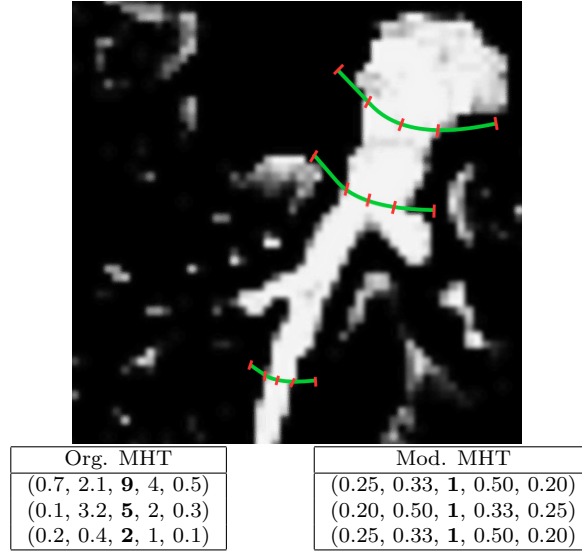


Figure 2.3: Illustration of a branch of varying dimensions along with the local hypothesis scores for the original and modified MHT methods. Notice how the scores of the best hypothesis at each scale varies for the original MHT method.

settings across scales. This behaviour is demonstrated for local hypothesis scores, computed using Eq. (2.4), in Figure 2.3. Specifically, local hypothesis scores tend to decrease with scale, making it challenging to obtain a single optimal local threshold that works well across scales. When the local threshold is tuned to extract smaller branches, then the chance of adding large and sub-optimal branches to the hypothesis tree is higher resulting in over-segmentation. Similarly, when the score is tuned for larger branches even the best candidate hypothesis for smaller branches might not qualify as valid candidates resulting in under-segmentation.

To alleviate the aforementioned scale dependence, we devise a strategy of ranking local hypotheses based on their scores. Instead of using the scores in Eq. (2.4) directly, we sort them in decreasing order to derive a ranking based score at each step in the range  $[0, 1]$ . If  $N$  local hypotheses are sampled at any given step, then the scores of the ranked local hypotheses are obtained as,

$$l'_i = \frac{1}{R_i} \quad \forall i \in 1 \dots N, \quad (2.6)$$

where  $R_i \in \{1, \dots, N\}$  is the rank of the  $N$  local hypotheses. By this scheme of relative scoring of hypotheses, where the best local hypothesis is assigned 1 at each tracking step, the problem of scale dependence of hypothesis scores

is immediately alleviated. This is demonstrated in the table to the right in Figure 2.3.

### Handling branching

Another factor to consider when tracking trees is branching. In the original MHT method, bifurcation detection is performed by clustering hypotheses based on their spatial location at every step using a spectral clustering algorithm [Chung, 1997]. However, there are no further details as to how the MHT tree is maintained after bifurcations. Once a bifurcation is detected, there are several ways to proceed. As the tracking happens per branch, new seed points can be added from the detected points of bifurcation and tracking can be restarted. This entails rebuilding of the MHT tree from each of the new seed points. The implementation of the original MHT method, available as a MeVisLab<sup>1</sup> module, follows this strategy. This has a negative consequence of discarding the information aggregated until the step before branching.

Another strategy that does not rely on rebuilding the MHT tree is to simply save the MHT tree at branching and resume tracking from each of the newly detected branching points separately. By resuming tracking from both branching points with the history of the parent branch we do not throw away information. In our modified MHT method, this strategy is used.

## 2.3 Experiments and Results

We evaluate the modified MHT method for two applications: extraction of airway trees and segmentation of coronary arteries. We used the MeVisLab implementation of the original MHT method for performing the original MHT experiments and a customised module with our modifications on MeVisLab for the modified MHT method. All experiments were performed on a laptop with 8 cores and 32 GB memory running Debian operating system. Details of preprocessing of data, experiments, error measures and results are presented next.

### 2.3.1 Airway tree extraction from lung CT

Experiments were performed on a subset of 32 low-dose CT scans from the Danish Lung Cancer Screening Trial [Pedersen et al., 2009] dataset. The 3D images in this dataset have a slice spacing of 1 mm and in-plane resolution varying between 0.72 to 0.78 mm. The images were randomly split into training

---

<sup>1</sup><http://mevislab.de/>

and test sets comprising of 16 images each. Performance of the method was compared with reference segmentations composed of an expert-corrected union of two previous methods: the first method uses an appearance model based on a voxel classifier to distinguish airway voxels from background and uses region growing along with a vessel similarity measure to extract airways [Lo et al., 2010], and the second method uses a similar voxel classifier but extracts airways by continually extending locally optimal paths [Lo et al., 2009]. The corrections performed by the human operator to obtain this reference provides us with a high quality reference to compare to.



Figure 2.4: Maximum intensity projection view of the probability image obtained from the voxel-classifier. Brighter regions correspond to higher probability, and hence more likely to belong to airways.

### Pre-processing of data

All CT images were pre-processed and converted into probability images using a kNN-based voxel classifier trained to distinguish airway voxels from background [Lo et al., 2010]. Thus obtained probability images have probability close to 1 in regions that are classified to be inside the airways and close to 0 outside; these images match the profile function described in Section 2.2, wherein, the structure of interest is bright (high probability) in a dark background. An example probability image is shown in Figure 2.4. Noise in the image is due to several factors including acquisition, interfering vessels, ribs and lung tissue.

Table 2.1: Optimal parameters for both MHT methods (Org. and Mod.) based on training set for airway extraction along with the search range of parameters. Parameters with \* were fixed based on the morphology of airways and *Search depth* was fixed based on initial experiments.

<sup>+</sup> We refer to the text in Section 2.3.1 for explanation on *Global threshold* for the original MHT method and two different ranges for *Weight window*.

Parameter/Method	Search Range	Org.	Mod.
Min. radius* (mm)	1	1	1
Max. radius* (mm)	10	10	10
Step length	[1.0, 1.1, ..., 2.0]	1.5	1.1
Weight window <sup>+</sup>	[3, 4, 5] / [1, 2, ..., 5]	3.0	1.0
Search depth	6	6	6
Search angle (deg.)	[30, 40, ..., 70]	60	70
Number of angles	[1, 2, ..., 5]	3	2
Local thres.( $T_{loc}$ )	[1.0, 2.0, ..., 5.0]	2.0	–
Global thres. <sup>+</sup>	$2T_{loc}$ / [0.5, 0.6, ..., 0.9]	4.0	0.7

### Error measure

Performance of all four methods are evaluated based on accuracy of the extracted centerlines. To standardise the evaluation procedure, centerlines were extracted from the binary segmentations from all methods using a 3D thinning algorithm [Lee et al., 1994]. The error measure is a symmetric distance between extracted and reference centerlines, given as,

$$d_{err} = w \frac{\sum_{i=1}^{n_{op}} \min d_E(c_i, C_{ref})}{n_{op}} + (1 - w) \frac{\sum_{j=1}^{n_{ref}} \min d_E(c_j, C_{op})}{n_{ref}}. \quad (2.7)$$

In the above equation,  $C_{ref}$ ,  $C_{op}$  are set of equidistant points on the centerlines of the reference and output segmentation results, respectively, comprising of  $n_{ref}$  and  $n_{op}$  number of points.  $c_i, c_j \in \mathbb{R}^n$  are individual points on the centerlines,  $d_E$  is the Euclidean distance and  $w$  is a weight, such that  $0 \leq w \leq 1$ . Notice that the first term in Eq. (2.7) captures the distance between the two centerlines due to false positives, whereas the second term captures the distance due to false negatives. In this work we use  $w = 0.5$ . Depending on the application the weight can be modified to obtain a desirable measure that reflects the sensitivity or specificity needs.

### Experiment set-up and parameter tuning

The modified MHT method was compared to the original MHT method and with region growing applied to both the probability and intensity images. Parameters of all the methods were tuned on the training set comprising 16 images and tested on an independent test set consisting of 16 images. Both the MHT methods have tunable parameters; the first set of parameters is related to the tubular template: minimum and maximum allowable radii of the templates, scaling factors for step length and weight window. A second category of parameters is related to multiple hypothesis tracking: search depth, search angle, number of angles, local and global hypothesis thresholds. Tuning both categories of parameters for the training set is cumbersome, so the minimum and maximum radii were fixed to 1mm and 10mm, respectively, based on prior knowledge about the morphology of airway trees. Also the search depth parameter was set to 6 for both methods based on initial experiments. Only the remaining parameters were tuned to minimise the training error defined in Eq. (2.7) using grid search. The range of parameters searched over and the optimal set of parameters obtained for both MHT methods are summarised in Table 2.1. Due to the scale independence introduced by the statistical ranking of local hypotheses, the modified MHT method does not have the local threshold as a parameter. The range of parameters for global threshold shown is for the modified MHT method, as the global threshold for the original MHT method was always set to be twice the optimal local threshold based on [Friman et al., 2010]. Both region growing methods only have the threshold parameter to be tuned. The optimal threshold for region growing on intensity of CT images was found to be  $-995$  HU which is a low threshold, due to the leakage for three images in the training set for any higher thresholds. The optimal threshold for the case of probability images based on the training set performance was found to be 0.5. All four methods require an initial seed point and the same seed point in the trachea, automatically extracted using the procedure described in [Lo et al., 2010], was provided to all.

We use the MHT module in MeVisLab provided by the authors in [Friman et al., 2010] to perform the original MHT experiments. In this implementation, minimum value for the weight window factor is constrained to be 3. However, for the modified MHT method we tried a larger range  $[1, 2, \dots, 5]$ , and found the training error to be slightly lower (by 0.2mm) when compared to using the smaller range  $[3, 4, 5]$ . We adhere to using this larger range for modified MHT method, and a smaller range for original MHT method, also for the Coronary Artery Extraction experiments.

Table 2.2: Average error comparison for extraction of airways

Method	$d_{err}$ (mm)
RG (intensity)	$3.368 \pm 0.808$
RG (probability)	$2.064 \pm 0.447$
Original MHT	$3.961 \pm 1.384$
<b>Modified MHT</b>	$1.929 \pm 0.494$

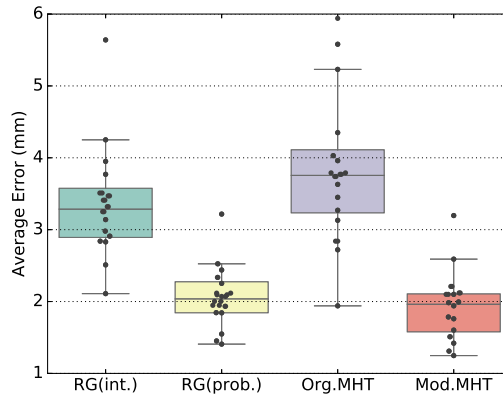


Figure 2.5: Average error on 16 scans in the test set to compare the modified MHT method with region growing (RG) on intensity images, region growing on probability images and the original MHT method, visualised as standard box plots. Both MHT methods were applied to the probability images.

## Results

Performance comparison was based on computing the average error measure in Eq. (2.7) and results for the test set are presented in Table 2.2 and visualised in Figure 2.5. Differences between methods were tested for statistical significance using paired sample  $t$ -tests. The modified MHT method clearly shows an improvement when compared to the original MHT method ( $p < 0.001$ ) and region growing on intensity images ( $p < 0.001$ ), but was not significantly better than region growing on probability images ( $p = 0.079$ ). Six of the test set results for the modified MHT method are visualised in Figure 2.6 along with the reference segmentations.

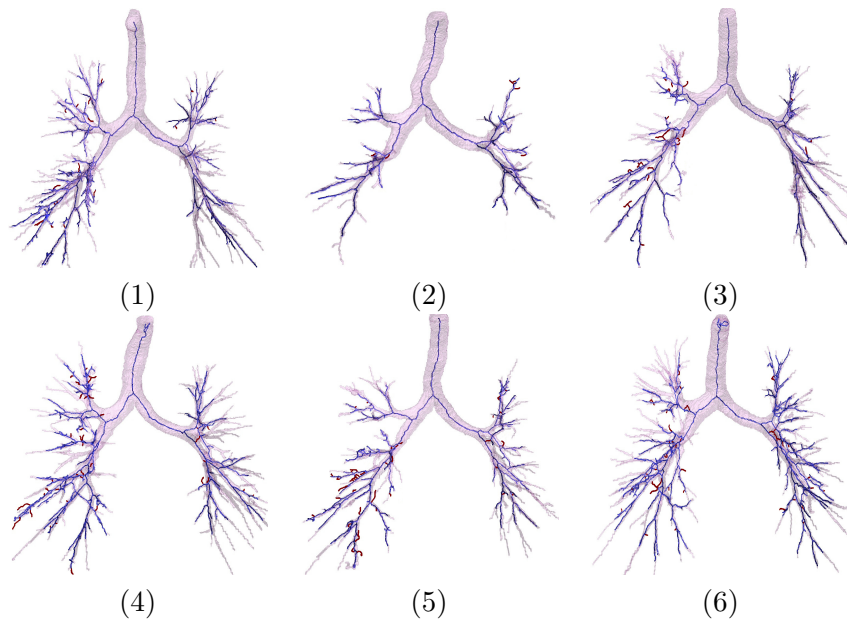


Figure 2.6: Six test case centerline results (thick lines) from the modified MHT method are shown overlaid on the reference segmentations (background surface). True positive portion of the extracted centerline is shown in blue and false positive portion in red.



### 2.3.2 Coronary Artery Extraction

We next describe experiments on 3D CT angiography (CTA) images for segmenting coronary arteries evaluated on the Coronary Artery Challenge data <sup>2</sup>. The challenge organisers allow methods to compete in three categories based on the extent of user interaction per vessel: automatic (no seed points), semi-automatic (one seed point) and interactive (more than one seed point). We evaluate the original and modified MHT methods as stand-alone, semi-automatic methods and compare to the results of [Friman et al., 2010].

#### Data and preprocessing

The dataset consists of 32 CTA images, split into 8 images for training and the remaining 24 for testing. These 3D volumes have a mean voxel size of  $0.32 \times 0.32 \times 0.4$  mm<sup>3</sup>.

The objective of the challenge is to segment four vessels per dataset. To this end, four seed points per vessel are provided, of which one must be selected for semi-automatic methods:

- Point S: start point of centerline
- Point E: end point of centerline
- Point B: a point about 3cm distal of the start point along the centerline
- Point A: a point inside distal part of the vessel.

In all images, three of the four vessels to be segmented were of the same type: right coronary artery (RCA), left anterior descending artery (LAD) and left circumflex artery (LCX). The fourth vessel, however, varied between images and was one of the large side branches (LSB). The reference standard for the training set was provided as manually drawn centerlines.

We closely adhere to the preprocessing performed in the original MHT method [Friman et al., 2008, Friman et al., 2010]. Unsigned integers are used to represent the voxel intensities as gray value (GV), under a simple transformation of the corresponding Hounsfield Unit (HU):  $HU(\mathbf{x}) = GV(\mathbf{x}) - 1024$ . To enable improved vessel template matching, voxel intensity of the lung tissue is raised to that of myocardial tissue and vessel calcifications are eliminated with the following thresholds:

$$I(\mathbf{x}) = \begin{cases} t_{myo} & \text{if } I(\mathbf{x}) < t_{myo} \\ I(\mathbf{x}) & \text{if } t_{myo} \leq I(\mathbf{x}) \leq t_{calc} \\ t_{myo} & \text{if } I(\mathbf{x}) > t_{calc}, \end{cases} \quad (2.8)$$

---

<sup>2</sup><http://coronary.bigr.nl/centerlines/index.php>

with  $t_{myo} = 950$  and  $t_{calc} = 1700$ .

### Error measure

We use two commonly reported measures on the challenge website in our evaluations, that quantify completeness and accuracy of the segmented vessels, defined as OV (overlap) and AI (average distance inside vessel) in [Schaap et al., 2009], respectively. We also report overlap until first error (OF) and overlap with the clinically part of the vessel (OT) to enable better comparison with other methods in the challenge.

Table 2.3: Optimal parameters for both MHT methods after tuning on the training set for Coronary Artery Challenge to maximise overlap (OV) along with the search range of parameters. Parameters with \* were fixed based on the morphology of vessels and *Search depth* was fixed based on initial experiments. <sup>+</sup> We refer to the text in Section 2.3.2 for explanation on *Global threshold* for the original MHT method and two different ranges for *Weight window*.

Parameter/Method	Search range	Org.	Mod.
Min. radius* (mm)	1	1	1
Max. radius* (mm)	3	3	3
Search depth	4	4	4
Weight Window <sup>+</sup>	[3, 4, 5] / [1, 2, ..., 5]	3	1
Step Length	[1.0, 1.1, ..., 2.0]	1.5	1.5
Search Angle (deg.)	[30, 40, ..., 70]	60	60
Number of angles	[1, 2, ..., 5]	3	2
Local Thres. ( $T_{loc}$ )	[1.0, 2.0, ..., 5.0]	4.0	–
Global Thres. <sup>+</sup>	$2T_{loc}$ / [0.55, 0.65, ..., 0.95]	8.0	0.95

### Experiment set-up and parameter tuning

As with the airway tree extraction experiments in Section 2.3.1, most of the parameters are tuned for both MHT methods, while some are fixed based on prior knowledge. The minimum and maximum radii parameters are set to 1mm and 3mm, respectively. *Search depth* was set to 4 based on initial experiments. Remaining parameters are tuned to maximise overlap, OV, on the training set and are summarised in Table 2.3. Further, as with the airway experiments, the starting value of weight window factor is 3, and the global threshold is twice the local threshold, for the original MHT method. Due to the varying quality of images across the training set, tracking from different seed points yields substantially different results. Based on the training starting tracking

from Point B and tracking in both directions turned out to be most useful for both MHT methods.

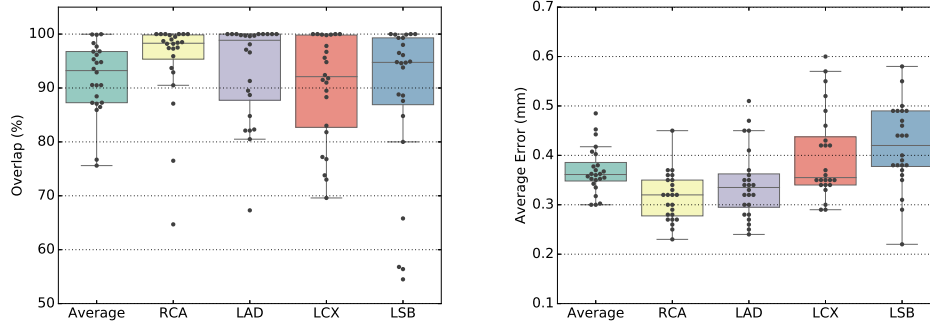


Figure 2.7: Overlap (OV) and accuracy (AI) measures for the modified MHT method evaluated on the coronary challenge test set.

## Results

We compare the modified MHT method with the original MHT method, evaluated as semi-automatic methods in the ongoing Coronary Artery Challenge evaluation. A version of the original MHT method, used along with a minimal path algorithm and user interaction [Friman et al., 2010] has remained the best performing method in the challenge. The modified MHT method scored 91.8% overlap and accuracy of 0.38mm, whereas the original MHT method obtained 67.6% overlap and accuracy of 0.39mm. There is significant improvement in overlap when compared to the original MHT method ( $p < 0.001$ ) as indicated by a paired sample  $t$ -test. With these results our method stands as the third best performing method amongst all semi-automatic methods (requiring one seed point) reported on the challenge website.

Test set measures for each of the four vessels, along with the average score per dataset, are shown in Figure 2.7 for the modified MHT method. The first plot in Figure 2.7 shows the overlap performance and it is clear that RCA, LAD and LCX vessels score high consistently and have few outliers. LSB, on the other hand, has four cases scoring less than 70% in OV. One possible reason is that the vessel chosen as LSB was different across the dataset and the tuned parameters could not generalise to these four cases. The second plot in Figure 2.7 depicts the accuracy (AI) for the test set across the four vessels and again the average error in LSB is slightly higher than the rest. In Table 2.4 we summarise the results for two versions of the original MHT method and the modified MHT

Table 2.4: Accuracy (AI) and different overlap measures (OV, OF, OT) for the original MHT method as interactive (int.) and semi-automatic (semi.) methods, and the modified MHT method.

Measure	MHT (int.)	MHT (semi.)	<b>Mod. MHT</b>
AI (mm)	$0.230 \pm 0.055$	$0.388 \pm 0.066$	$0.368 \pm 0.047$
OV (%)	$98.521 \pm 02.183$	$67.612 \pm 13.334$	$91.783 \pm 06.630$
OF (%)	$83.078 \pm 28.765$	$44.078 \pm 21.326$	$70.992 \pm 21.327$
OT (%)	$98.681 \pm 03.557$	$70.034 \pm 13.289$	$93.372 \pm 06.072$

method.

## 2.4 Discussion and Conclusions

In this work, we extended the template-matching based method for extraction of elongated structures using multiple hypothesis tracking in [Friman et al., 2010]. We introduced statistical ranking of local hypotheses as a possible means to overcome scale dependence of important MHT parameters.

A useful interpretation of the proposed ranking scheme is to see it as a measure of the relative significance of local hypotheses at each step. When aggregated over the depth of the hypothesis tree, it can be interpreted as the likelihood of the corresponding global hypothesis. Consider an example when the search depth is 10 and the global hypothesis threshold is 0.6. For a global hypothesis (comprising of a sequence of 10 local hypotheses) to score above 0.6 it would require its component local hypotheses to have been best locally at least in 6 of the 10 tracking steps, on average. This relative significance provides a probabilistic global hypothesis score, making it independent of the scale and variations of structures and removes the need to perform two level pruning of hypotheses – using local and global hypothesis thresholds – as done in the original MHT method. Instead of explicitly pruning local hypotheses, we propagate all local hypotheses with their ranked scores from Eq. (2.6) and rely on the deferred decisions of MHT algorithm to yield valid branches.

One possible concern that arises when the scores in Eq. (2.4) are discarded for relative ranking in Eq. (2.6) is the chance of poor candidate hypotheses performing well when ranked amongst equally poor counterparts. A strength of the deferred decision of MHT is that with adequate depth of the MHT tree such hypotheses are unlikely to be accepted, as they would have to perform well across multiple steps. Increasing search depth increases the computational

expense as the number of hypotheses can grow exponentially. With both our experiments we observed that search depth of upto 6 is feasible to run on limited computational resources, yielding promising results.

Due to these changes the modified MHT method can now be considered automatic, requiring only one seed point per tree. In most cases, even the one required seed point can also be automatically provided, as in the case of airway experiments in Section 2.3.1, and thus making the method fully automatic. We also presented an improvement to the bifurcation handling strategy by maintaining the history of the MHT tree across bifurcations instead of restarting tracking at bifurcation points.

We presented evaluations in Section 4.3, where we first focussed on extracting airway trees from chest CT data. To the extent of our knowledge, this is the first application of an MHT-based method to the task of airway extraction. The modified MHT method was compared with region growing on intensity images, region growing on probability images and the original MHT method. The modified MHT method shows significant improvement in average centerline distance when compared to the original MHT method ( $p < 0.001$ ). The modified MHT method required 15 seconds on average, per CT scan, to extract airway trees starting from a single seed point.

We also presented evaluations on the Coronary Challenge [Schaap et al., 2009] and compared the modified MHT method with the original MHT method used as a semi-automatic method. The interactive version of the original MHT method, which is a combination of MHT, minimal paths and extensive user interaction (on average 1.5 clicks per vessel), is still the best performing method in the Coronary Artery Challenge. The modified MHT method, however, shows significant improvement in the overlap measure (OV) when compared to the semi-automatic original MHT method ( $p < 0.001$ ). The accuracy of the extracted centerlines (AI) from both semi-automatic versions of the MHT method are close, indicating that the original MHT method (used semi-automatically) was accurate in the vessels that were segmented, but it under-segments the vessels indicated by the low OV score. This is likely due to dependence of the parameters on scale, which is overcome with the statistical ranking introduced in the modified MHT method. In this work we evaluated the modified MHT method using one seed point per vessel for the gains in some of the cases, while still competing in the semi-automatic category of the challenge. However, the modified MHT method could also have been used with fewer than one seed point per vessel (one seed point per tree) and yielding similar performance, as we were able to extract substantial portions of the coronary tree starting from a single seed point. The computation time required for the modified MHT method per vessel is 3 seconds on average; thus not more than 15 seconds per CTA image in the dataset. This

makes it more appealing for clinical usage than the original MHT method that similar computation time per time plus the time used for user interaction.

There are two recent semi-automatic methods based on diffusion tensor imaging techniques [Cetin et al., 2013, Cetin and Unal, 2015] that perform better than the modified MHT method in the OV measures as reported in the challenge website with a score of 97.3% and 96.4%, respectively. However, in the OF measure these methods score 69.9% and 77.4%, respectively, while the modified MHT method obtained 77.9%. Also, the accuracy measure is comparable for all three methods ( $\approx 0.35$  mm). Three other methods in the fully automatic category have better OV measure by a small margin, between 1 – 2%, when compared to the modified MHT method [Yang et al., 2012, Zheng et al., 2013, Bauer and Bischof, 2008]. Again, the modified MHT method obtains 3% higher OF score when compared to these methods. One common feature across all these better performing methods is the pre-processing performed on the CT images to enhance vessel-like structures. For instance, in the currently best performing fully automatic method an improved version of Frangi’s vesselness is used before centerline extraction [Yang et al., 2012], and similarly in [Bauer and Bischof, 2008] a tube detection filter is used and a pre-trained vessel detection filter is used in [Zheng et al., 2013]. In the modified MHT method we use a simple template-matching step and, use of more sophisticated strategies as described in the other methods in the challenge should further improve the performance. For instance, we expect that the use of probability images obtained from a voxel classifier to distinguish vessel and non-vessel voxels instead of intensity images, as in the case of the airway experiments, could have been beneficial.

In conclusion, we proposed modifications to the well established original MHT method that yield significantly better results. Statistical ranking of local hypotheses yields a common interpretation of the global hypotheses of the MHT tree at all scales. The proposed modifications also make it possible to use the method in a non-interactive manner and yield competitive results on versatile datasets involving segmentation of complete tree structures.

## Acknowledgements

This work was funded by the Independent Research Fund Denmark (DFF) and Netherlands Organisation for Scientific Research (NWO).

## Chapter 3

# Bayesian Smoothing

The work presented in this chapter is based on Raghavendra Selvan, Jens Petersen, Jesper H. Pedersen, and Marleen de Bruijne. “Extraction of airways with probabilistic state-space models and Bayesian smoothing.” In *Graphs in Biomedical Image Analysis, Computational Anatomy and Imaging Genetics*, MICCAI, 2017, pp. 53-63. Springer, Cham.

## Abstract

Segmenting tree structures is common in several image processing applications. In medical image analysis, reliable segmentations of airways, vessels, neurons and other tree structures can enable important clinical applications. We present a framework for tracking tree structures comprising of elongated branches using probabilistic state-space models and Bayesian smoothing. Unlike most existing methods that proceed with sequential tracking of branches, we present an exploratory method, that is less sensitive to local anomalies in the data due to acquisition noise and/or interfering structures. The evolution of individual branches is modelled using a process model and the observed data is incorporated into the update step of the Bayesian smoother using a measurement model that is based on a multi-scale blob detector. Bayesian smoothing is performed using the RTS (Rauch-Tung-Striebel) smoother, which provides Gaussian density estimates of branch states at each tracking step. We select likely branch seed points automatically based on the response of the blob detection and track from all such seed points using the RTS smoother. We use covariance of the marginal posterior density estimated for each branch to discriminate false positive and true positive branches. The method is evaluated on 3D chest CT scans to track airways. We show that the presented method results in additional branches compared to a baseline method based on region growing on probability images.

## 3.1 Introduction

Segmentation of tree structures comprising of vessels, neurons, airways etc. are useful in extraction of clinically relevant biomarkers [Lesage et al., 2009, Lo et al., 2012]. The task of extracting trees, mainly in relation to vessel segmentation, has been studied widely using different methods. A successful class of these methods are based on techniques from target tracking. Perhaps the most used tracking strategy is to proceed from an initial seed point, make local-model fits to track individual branches in a sequential manner and perform regular branching checks [Friman et al., 2008, Yedidya and Hartley, 2008]. Such methods are prone to local anomalies and can prematurely terminate if occlusions are encountered. The method in [Friman et al., 2008] can overcome such problems to a certain extent using a deterministic multiple hypothesis testing approach; however, it is a semi-automatic method requiring extensive manual intervention and can be computationally expensive. In [Yedidya and Hartley, 2008], vessel tracking on 2D retinal scans is performed using a Kalman filter. They propose an automatic seed point detection strategy using a matched filter. From each of these seed points vessel branches are progressively tracked using measurements that are derived



from the image data. A gradient based measurement function is employed which fails in low-contrast regions of the image, which are predominantly regions with thin vessels. Another major class of tracking algorithms are based on a stochastic formulation of tracking [Florin et al., 2005, Lesage et al., 2016] using some variation of particle filtering. Particle filter-based methods are known to scale poorly with dimensions of the state space [Lesage et al., 2009].

In spirit, we propose an exploratory method like particle filter-based methods, with a salient distinction that the proposed method can track branches from several seed points across the volume. We use linear Bayesian smoothing to estimate branch states, described using Gaussian densities. Thus, the method inherently provides an uncertainty measure, which we use to discriminate true and false positive branches. Further, unlike particle filter-based methods, the proposed method is fast, as Bayesian smoothing is implemented using the RTS (Rauch-Tung-Striebel) smoother [Särkkä, 2013] involving only a set of linear equations.

## 3.2 Method

We formulate tracking of branches in tree structures using probabilistic state-space models, commonly used in target tracking and control theory [Särkkä, 2013]. The proposed method takes image data as input and outputs a collection of disconnected branches that taken together forms the tree structure of interest. We first process the image data to obtain a sequence of measurements and track all possible branches individually using Bayesian smoothing. We then use covariance estimates of individual branches to output a subset of the most likely branches yielding the tree structure of interest. Details of this process are described below.

### Tracking individual branches

We assume the tree structure of interest,  $\mathbf{X}$ , to be a collection of  $T$  independent random variables  $\mathbf{X} = \{\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_T\}$ , where individual branches are denoted  $\mathbf{X}_i$ . Each branch  $\mathbf{X}_i$  of length  $L_i$  is treated as a sequence of states,  $\mathbf{X}_i = [\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_{L_i}]$ . These states are assumed to obey a first-order Markov assumption, i.e.,

$$p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{x}_{k-2}, \dots, \mathbf{x}_0) = p(\mathbf{x}_k | \mathbf{x}_{k-1}). \quad (3.1)$$

The state vector has seven random variables,

$$\mathbf{x}_k = [x, y, z, r, v_x, v_y, v_z]^T, \quad (3.2)$$

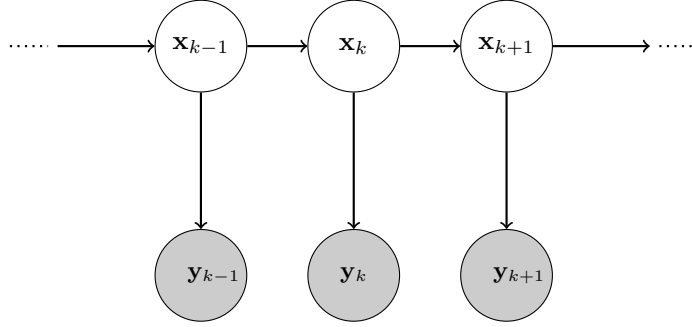


Figure 3.1: Bayesian network view of the relation between the underlying true states,  $\mathbf{x}_i$ , and the measurements,  $\mathbf{y}_i$ , for a single branch.

describing a tubular segment centered at Euclidean coordinates  $[x, y, z]$ , along an axis given by the direction vector  $[v_x, v_y, v_z]$  with radius  $r$ .

The observed data, image  $\mathbf{I}$ , is processed to be available as a sequence of vectors. We model the measurements as four dimensional state vectors consisting only of position and radius. This is accomplished using a multi-scale blob detector [Lindeberg, 1998]. The input image  $\mathbf{I}$  with  $N_v$  voxels is transformed into a sequence of  $N$  measurements, with position and radius information, denoted  $\mathbf{Y} = [\mathbf{y}_0, \dots, \mathbf{y}_N]$ , where each  $\mathbf{y}_i = [x, y, z, r]^T$ . This procedure applied to the application of tracking airway trees is described in Section 3.2.

### Process and Measurement Models

Transition from one tracking step to another within a branch is modelled using the process model. We use a process model that captures our understanding of how individual branches evolve between tracking steps and has similarities with the model used in [Yedidya and Hartley, 2008]. We assume first-order Markov independence in state transitions from (3.1), captured in the process model below:

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{q} = \begin{bmatrix} 1 & 0 & 0 & 0 & \Delta & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & \Delta & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & \Delta \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ z_{k-1} \\ r_{k-1} \\ v_{x_{k-1}} \\ v_{y_{k-1}} \\ v_{z_{k-1}} \end{bmatrix} + \mathbf{q} \quad (3.3)$$

where  $\mathbf{F}$  is the process model function and  $\mathbf{q}$  is the process noise.  $\mathbf{q}$  is assumed to be a zero mean Gaussian density, i.e,  $\mathbf{q} \sim N(\mathbf{0}, \mathbf{Q})$ , with process covariance,  $\mathbf{Q}_{7 \times 7}$ , acting only on direction and radius components of the state vector,

$$\mathbf{Q}_{[4:7,4:7]} = \sigma_q^2 \Delta \times \mathbf{I}_{4 \times 4}, \quad (3.4)$$

where only the non-zero part of the matrix is shown and  $\sigma_q^2$  is the process variance. The parameter  $\Delta$  can be seen as step size between tracking steps. As (3.3) is a recursion, the initial point (seed point),  $\mathbf{x}_0$ , comprising of position, scale and orientation information is provided to the model. Seed points are assumed to be described by Gaussian densities,  $\mathbf{x}_0 \sim N(\hat{\mathbf{x}}_0, \mathbf{P}_0)$ , with mean  $\hat{\mathbf{x}}_0$  and covariance  $\mathbf{P}_0$ . We present an automatic strategy to detect such initial seed points in 3.2.

The measurement model describes the relation between each of the 4-D measurements,  $\mathbf{y}_k$  in the sequence,  $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_N]$ , and the state vector,  $\mathbf{x}_k$ , as shown in Figure 3.1. A simple linear measurement model captures this relation,

$$\mathbf{y}_k = \mathbf{H}\mathbf{x}_k + \mathbf{m} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ z_k \\ r_k \\ v_{x_k} \\ v_{y_k} \\ v_{z_k} \end{bmatrix} + \mathbf{m} \quad (3.5)$$

where  $\mathbf{y}_k$  are observations generated by true states of the underlying branch at step  $k$ ,  $\mathbf{H}$  is the measurement function.  $\mathbf{m} \sim N(\mathbf{0}, \mathbf{R})$  is the measurement noise with covariance  $\mathbf{R}$  that is a diagonal matrix with entries,  $[\sigma_{m_x}^2, \sigma_{m_y}^2, \sigma_{m_z}^2, \sigma_{m_r}^2]$ , which correspond to variance in the observed position and radius, respectively. All possible measurement vectors obtained from the image are aggregated into the measurement variable  $\mathbf{Y}$ .

### Bayesian Smoothing

The state-space models presented above enable us to estimate branches using the posterior distributions,  $p(\mathbf{X}_i | \mathbf{Y}) \forall i = [0, \dots, T]$ , using standard Bayesian methods. We employ Bayesian smoothing as all the measurements are available at once, when compared to sequential observations that are more common in object tracking applications. Due to a linear, Gaussian process and measurement models, Bayesian smoothing can be optimally performed using the RTS smoother [Särkkä, 2013]. RTS smoother uses two Bayesian filters to perform forward filtering and backward smoothing. Forward filtering is identical to

Table 3.1: Standard RTS Smoother Equations

<b>Forward Filtering</b>	<b>Backward Smoothing</b>
$\hat{\mathbf{x}}_{k k-1} = \mathbf{F}\hat{\mathbf{x}}_{k-1 k-1}$ (3.6)	
$\mathbf{P}_{k k-1} = \mathbf{F}\mathbf{P}_{k-1 k-1}\mathbf{F}^T + \mathbf{Q}$ (3.7)	$\mathbf{G}_k = \mathbf{P}_{k k}\mathbf{F}^T\mathbf{P}_{k+1 k}^{-1}$ (3.13)
$\mathbf{v}_k = \mathbf{y}_k - \mathbf{H}\hat{\mathbf{x}}_{k k-1}$ (3.8)	$\hat{\mathbf{x}}_{k L} = \hat{\mathbf{x}}_{k k} + \mathbf{G}_k(\hat{\mathbf{x}}_{k+1 L} - \hat{\mathbf{x}}_{k+1 k})$ (3.14)
$\mathbf{S}_k = \mathbf{H}\mathbf{P}_{k k-1}\mathbf{H}^T + \mathbf{R}$ (3.9)	
$\mathbf{K}_k = \mathbf{P}_{k k-1}\mathbf{H}^T\mathbf{S}_k^{-1}$ (3.10)	$\mathbf{P}_{k L} = \mathbf{P}_{k k} - \mathbf{G}_k(\mathbf{P}_{k+1 k} - \mathbf{P}_{k+1 L})\mathbf{G}_k^T$ (3.15)
$\hat{\mathbf{x}}_{k k} = \hat{\mathbf{x}}_{k k-1} + \mathbf{K}_k\mathbf{v}_k$ (3.11)	
$\mathbf{P}_{k k} = \mathbf{P}_{k k-1} - \mathbf{K}_k\mathbf{S}_k\mathbf{K}_k^T$ (3.12)	

performing Kalman filtering and consists of sequential prediction and update with observed information of the state variable. Once a branch is estimated using forward filtering, the saved states are used to perform backward smoothing using a Kalman-like filter which improves state estimates by incorporating additional information from future steps. Standard equations for an RTS smoother are presented below [Särkkä, 2013].

### Forward Filtering

Equations in the first column of Table 3.1 are used to perform prediction and update steps of the forward filtering. In the prediction step, process model is used to predict states at the next step. Mean  $\hat{\mathbf{x}}_{k|k-1}$  and covariance  $\mathbf{P}_{k|k-1}$  estimates of the predicted Gaussian density, i.e, of state  $k$  conditioned on the previous state, denoted with subscript  $k|k-1$ , are computed in (3.6),(3.7). In the update step, described in (3.8) – (3.12), predicted density is associated with a measurement vector to obtain posterior density. First, the new information from measurement  $\mathbf{y}_k$  is computed using (3.8) and is aptly called the “innovation”, denoted as  $\mathbf{v}_k$ . Uncertainty in the new information, innovation covariance  $\mathbf{S}_k$ , is computed in (3.9). Then, predicted mean is adjusted with weighted innovation and predicted covariance is adjusted with weighted innovation covariance to obtain the posterior mean and covariances, in (3.11) and (3.12), respectively. The weighting computed in (3.10), denoted as  $\mathbf{K}_k$ , is the Kalman gain which controls the extent of information fusion from process and measurement models.

We continue estimation of the posterior density (described by posterior mean and covariance) in a sequential manner for the branch until no new measurements

exist for updating. After the final update step, a sequence of posterior mean estimates  $[\hat{\mathbf{x}}_{0|0}, \dots, \hat{\mathbf{x}}_{L_i|L_i}]$  and posterior covariance estimates  $[\mathbf{P}_{0|0}, \dots, \mathbf{P}_{L_i|L_i}]$ , obtained from the forward filter are saved, for further use by the backward smoother.

### Backward smoothing

The smoothed estimates are obtained by running a backward filter starting from the final tracked state of the forward filter. The intuition behind backward smoothing is that the uncertainty in making predictions in the forward filtering can be alleviated using information from future steps. It is implemented using the equations in the second column of Table 3.1.

### Gating

When performing the RTS smoother recursions, the forward filter expects a single measurement vector for the update step. We employ rectangular and ellipsoidal gating to reduce the number of measurements handled during the update step [Bar-Shalom et al., 2011].

First, we perform simple rectangular gating which is based on excluding measurements that are outside a rectangular region around the predicted measurement  $\mathbf{H}\hat{\mathbf{x}}_{k|k-1}$  in equation (3.8) using the following condition:

$$|\mathbf{y}_i - \mathbf{H}\mathbf{x}_{k|k-1}| \leq \kappa \times \text{diag}(\mathbf{S}_k), \forall \mathbf{y}_i \in \mathbf{Y} \quad (3.16)$$

where  $\mathbf{S}_k$  is the covariance of the predicted measurement in equation (3.9). The rectangular gating coefficient,  $\kappa$ , is usually set to a value  $\geq 3$  [Bar-Shalom et al., 2011]. Rectangular gating localises the number of candidate measurements relevant to the current tracking step. To further narrow down on the best candidate measurement for update, we follow rectangular gating with ellipsoidal gating [Bar-Shalom et al., 2011]. With ellipsoidal gating we accept the measurements within the ellipsoidal region of the predicated covariance, using the following rule:

$$(\mathbf{H}\mathbf{x}_{k|k-1} - \mathbf{y}_i)^T \mathbf{S}_k^{-1} (\mathbf{H}\mathbf{x}_{k|k-1} - \mathbf{y}_i) \leq G \quad (3.17)$$

where  $G$  is the rectangular gating threshold, obtained from the gating probability  $P_g$ , which is the probability of observing the measurement within the ellipsoidal gate,

$$P_g = 1 - \exp\left(-\frac{G}{2}\right). \quad (3.18)$$

### Tree as a Collection of Branches

Once a branch is smoothed and saved using Bayesian smoothing described previously, we process new seed points and start tracking branches until no further seed points remain to track from. This procedure yields a collection of disconnected branches. The next task is to obtain a subset of likely branches that represent the tree structure of interest by discarding false positive branches.

### Validation of Tracked Branches

An advantage of using Bayesian smoothing to track individual branches is that apart from estimating the branch states from the image data (using the smoothed posterior mean estimates), we can also quantify the uncertainty of the estimation at each tracking step (using the smoothed posterior covariance estimates). Thus, we have the possibility of aggregating this uncertainty over the entire branch to validate them. We explore this notion to create a criterion for accepting or rejecting branches.

By aggregating variance for all tracking steps in each branch, we obtain a measure of the quality of branches. A straightforward approach is to use total variance, obtained using the trace of each of the smoothed posterior covariance matrices. We average the sum total variance over the length of each branch,  $l_i$ , to obtain a score,  $\mu_i$ , which is then thresholded by a cut-off  $\mu_c$  to qualify the branches,

$$\mu_i = \frac{\sum_{k=1}^{l_i} \text{Tr}(\mathbf{P}_{k|k})}{l_i}. \quad (3.19)$$

### Application to Airways

The proposed method for tracking tree structures can be applied to track airways, vessels or other tree structures encountered in image processing applications. We focus on tracking airways from lung CT data and present the specific strategies used to implement the proposed method.

### Multi-scale representation

The measurement model discussed in Section 3.2 assumes a 4-D state vector as measurements to the RTS smoother. This is achieved by first computing an airway probability image using a k-Nearest Neighbour voxel classifier trained to discriminate between airway and background, described in [Lo et al., 2010]. Blob detection with automatic scale selection [Lindeberg, 1998] for different scales,  $\sigma_s = (1, 2, 4, 8, 12)mm$ , is performed on the probability image to obtain the 4D state measurements as blob position and radius. Indistinct blobs are

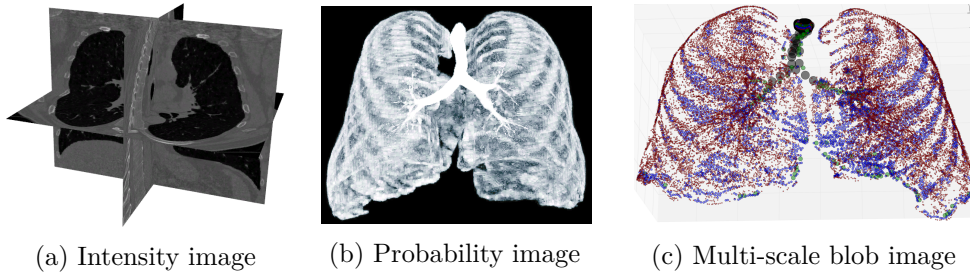


Figure 3.2: The pipeline of image representations, ultimately showing the multi-scale representation.

removed if the absolute value of the normalized response at the selected scale,  $\sigma_s^*$ , is less than a threshold [Lindeberg, 1998]. This makes the representation sparse,  $N \ll N_v$ , and the tracking more efficient than if performed at voxel-level. An example of the sparse representation can be found in Figure 3.2.

### Initialisation of Branches

The multi-scale representation of the image data discussed above also provides a response corresponding to the best scale. As this response is normalised for scales, we incorporate this information in selecting the initial seed point for every branch. We start tracking from the seed point with the largest scale and the largest response. The initial direction information is obtained from eigen value analysis of the Hessian matrix computed at the corresponding scale provided in the measurement vector. Once a branch is tracked along the initial direction, we track from the same seed point but in the opposite direction. Thus, if a seed point is obtained from the middle of a branch we can track it bidirectionally. After tracking in both directions, all the involved measurements including the seed point are removed from the measurement vector, and the next best candidate seed point is chosen and tracking commences from there. The tracking procedure on the entire image is complete when no more seed points are available.

## 3.3 Experiments and Results

### Data

The evaluation was carried out on 32 low-dose CT chest scans from a lung cancer screening trial [Pedersen et al., 2009]. Training and test sets comprising

of 16 images each were randomly obtained from the data set. All scans have a resolution of approximately  $1\text{mm} \times 0.78\text{mm} \times 0.78\text{mm}$ . The reference segmentations consist of expert verified union over the results of two previous methods [Lo et al., 2009, Lo et al., 2010]. The proposed method is compared with region growing on the probability images.

### Error Measure, Initial Parameters and Tuning

We use an error measure defined as the average of two distances,  $d_{err} = (d_{FP} + d_{FN})/2$ . The first distance,  $d_{FP}$ , captures the false positive error and is the average minimum Euclidean distance from segmentation centerline points to reference centerline points.  $d_{FN}$  similarly defines the false negative error, as the average minimum Euclidean distance from reference centerlines points to segmentation centerline points.

There are several parameters related to the RTS smoother that need to be initialised. These parameters were tuned using the training set and fixed for the evaluation on the test set to: standard deviations of the process noise,  $\sigma_q = 0.3$ , measurement noise on radius  $\sigma_{m_r} = 1$  mm and measurement noise on position  $(\sigma_{m_x}, \sigma_{m_y}, \sigma_{m_z}) = 2$  mm. The initial covariance,  $\mathbf{P}_0$  across branches was set to  $\mathbf{I}_{7 \times 7}$ . The most crucial parameter in the proposed method is the threshold parameter  $\mu_c$  presented in Section 3.2. The threshold to validate branches is tuned to be  $\mu_c = 2.0$ . The gating probability was set to a high value,  $P_g = 0.99$  [Bar-Shalom et al., 2011].

### Results

Figure 3.3 illustrates features of the proposed method by visualising centerlines overlaid on the reference segmentation. Influence of the threshold parameter  $\mu_c$  is illustrated with the segmentation results for a single volume without any threshold (seen in Figure 3.3a) and after applying the tuned threshold (seen in Figure 3.3b). Evidently, thresholding the average total variance of a branch eliminates false positive branches.

The final output obtained from the method is a collection of disconnected branches. While such collection of branches are still useful in extracting biomarkers, for evaluation purposes we merge the results obtained with the segmentations from region growing on probability images and extract centerlines from the merged segmentation using 3D thinning, as seen in Figure 3.3a and 3.3b. This also allows us to demonstrate the improvement our method provides by extracting peripheral airway branches, which are typically the challenging ones. One such combined result is shown in Figure 3.3c, where the yellow centerlines correspond to region growing and blue one is the combined result.



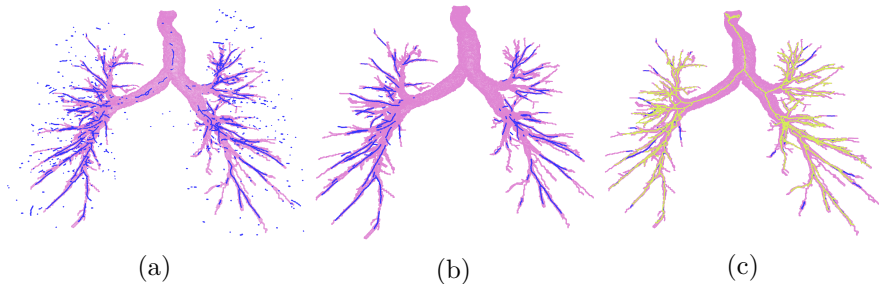


Figure 3.3: Visualisation of the centerlines extracted using the proposed method before and after thresholding to discard false positive branches overlaid on the reference segmentation, shown in (a) and (b) respectively. The combined results from the proposed method and region growing on probability is shown as the blue centerline in (c).

Table 3.2: Performance comparison on the test set

Method	$d_{FP}$ (mm)	$d_{FN}$ (mm)	$d_{err}$ (mm)	Std.Dev. (mm)
RG	0.423	3.579	2.001	0.208
(RTS+RG) <sub>1</sub>	0.449	2.102	1.276	0.187
(RTS+RG) <sub>2</sub>	0.401	2.658	1.529	0.165

Performance on the test set for two different scenarios of the proposed method is reported in Table 4.1 along with the numbers for region growing on probability images. The result for the best performing region growing on probability images is denoted with RG and those obtained by combining the proposed method with region growing are denoted as RTS+RG. We first combine the proposed method with the best performing region growing case (with minimum  $d_{err}$ ) results and it is denoted as (RG+RTS)<sub>1</sub>. We observe an improvement of about 36% on  $d_{err}$ . It is to be noted, there is substantial reduction in  $d_{FN}$ , indicating that many branches missed by region growing are now segmented. There is a very small increase in false positives which could also be due to the missing branches in the reference segmentation; however, the net result is a large improvement. To test whether the proposed method can simultaneously reduce the number of false positives and false negatives compared to region growing, we merge the proposed method with the region growing result that yields non-optimal  $d_{err}$ , and do observe a reduction in both  $d_{FP}$  and  $d_{FN}$  when compared to the best performing RG as seen in the entries for (RG+RTS)<sub>2</sub>.

The computational expense for running the proposed method is small. The

largest chunk of it is used in generating the multi-scale representation of the images, which is in the range of 10-15s per volume. Tracking using the RTS smoother and obtaining the segmentation takes about 4s on a laptop with 8 cores and 32 GB memory running Debian operating system.

### 3.4 Discussion and Conclusions

We presented an automatic method for tracking tree structures, in particular airways, using probabilistic state-space models and Bayesian smoothing. We demonstrated that branches can be tracked individually from across the volume, starting from several seed points. This approach of tracking branches from across the volume has the advantage that even in the presence of occlusions, such as mucous plugging or image acquisition noise, the chances of detecting branches beyond the occlusions are higher. An inherent measure of uncertainty in the branch estimates has been presented due to the Bayesian nature of the method. We demonstrated the use of thresholding this uncertainty measure to discriminate detected branches. The use of sparse representation of voxels in the image using blob detection makes the method computationally efficient.

A possible limitation with the proposed method is that it yields a disconnected tree structure. For applications where this is an issue, one can enforce a global connectivity constraint on the disconnected set of branches to obtain fully connected tree as done in [Graham et al., 2010] or similar. It is also possible to derive biomarkers directly from the disconnected branches, as shown in [Sørensen et al., 2011].

We performed an evaluation of the results obtained from the proposed method by combining it with the results from region growing on probability images. We showed that there is substantial improvement in the segmentation results, indicating that the exploratory approach taken up in our method has potential in improving tree segmentations.

### Acknowledgements

This work was funded by the Independent Research Fund Denmark (DFR) and Netherlands Organisation for Scientific Research (NWO).

## Chapter 4

# Graph Refinement Methods

The work presented in this chapter is based on a paper that is to be submitted as [1]. Portions of the work in this chapter are based on [2,3].

- [1] Raghavendra Selvan, Thomas Kipf, Max Welling, Jesper H. Pedersen, Jens Petersen, and Marleen de Bruijne. “Extracting Tree Structures using Mean-Field Networks and Graph Neural Networks” (2018)
- [2] Raghavendra Selvan, Max Welling, Jesper H. Pedersen, Jens Petersen, and Marleen de Bruijne. “Mean field network based graph refinement with application to airway tree extraction.” 21st Conference on Medical Image Computing & Computer Assisted Intervention (MICCAI 2018), pp. 750-758, Cham. Springer International Publishing.
- [3] Raghavendra Selvan, Thomas Kipf, Max Welling, Jesper H. Pedersen, Jens Petersen, and Marleen de Bruijne. “Extraction of Airways using Graph Neural Networks.” 1st Conference on Medical Imaging with Deep Learning (MIDL 2018), Amsterdam.

## Abstract

Graph refinement, or the task of obtaining subgraphs of interest from over-complete graphs, can have many varied applications. In this work, we focus on extraction of tree structures from image data by, first deriving graph like representation of the volumetric data and then, posing it as a graph refinement task. We present two methods to perform graph refinement. First, we use mean-field approximation (MFA) to approximate the posterior density over the subgraphs from which the optimal subgraph of interest can be estimated. Mean field networks (MFNs) are used for inference based on the interpretation that iterations of MFA can be seen as feed-forward operations in a neural network. This allows us to learn the model parameters using gradient descent. Second, we present a supervised learning approach using graph neural networks (GNNs) which can be seen as generalisations of MFNs. Subgraphs are obtained by jointly training a GNN based encoder-decoder pair to learn useful edge embeddings, from which the edge probabilities are predicted using a simple decoder. We discuss the connections between the two classes of methods and compare them qualitatively for the task of extracting airways from 3D, low-dose, chest CT data. We show that both the MFN and GNN models show significant improvement when compared to the baseline method in detecting more branches.

*Index terms* – mean-field networks, graph neural networks, tree segmentation, CT, airways

## 4.1 Introduction

Segmenting tree structures from image data is a commonly encountered image analysis task. In particular, there are several naturally occurring trees in medical images, and extracting them automatically can be of immense clinical value. Airways, vessels and neurons are some such structures that have been studied extensively from a segmentation point of view [Lesage et al., 2009, Donohue and Ascoli, 2011, Pu et al., 2012, Lo et al., 2012].

Graph based methods have been previously used for the extraction of vessels [Orlando and Blaschko, 2014], airways [Bauer et al., 2015] and neurons [Türetken et al., 2016]. In [Orlando and Blaschko, 2014] a conditional random field (CRF) based model is presented, with parameterised node and pairwise potentials for the segmentation of 2D retinal blood vessels. The parameters of the CRF are learned from training data using support vector machines. In [Bauer et al., 2015], a tube detection filter is used to obtain candidate airway branches. These candidate branches are represented as nodes in a graph and airway trees are reconstructed using a two-step graph based optimisation proce-

dure. In [Türetken et al., 2016] the image data is first processed to obtain local regions of interest using a path classifier and these regions are over-connected to obtain a graph based representation. Then, structures of interest like vessels and axonal trees are reconstructed from this graph using integer linear programming.

In this work, we take up a similar approach by formulating extraction of tree structures from volumetric image data as a graph refinement task. The image data is first processed to obtain graph-like representations, comprising nodes with expressive local features. This input graph is initially over-connected based on simple neighbourhood criteria and then refined to obtain the optimal subgraph. One of the motivations for posing tree extraction as a graph refinement task is to present an exploratory method to segment tree-structures.

Many widely used methods for vascular and airway tree segmentations are sequential in nature i.e, they start from one location (a seed point) and segment trees by making successive local decisions [Lesage et al., 2009, Pu et al., 2012, Lo et al., 2009, Lo et al., 2010, Selvan et al., 2018b]. For instance, in the EXACT’09 airway segmentation challenge [Lo et al., 2012], 10 out of the 15 competing methods used some form of region growing to make the segmentation decisions and the remainder of the methods also were sequential. Methods that rely on sequential and local decisions are susceptible to local anomalies in the data due to noise and/or occlusion and can possibly miss large portions of the tree-structures. We present the graph refinement approach in this work to address these concerns.

We propose two approaches to solve the graph refinement task, for extracting tree-like structures, using : 1) Mean-Field Networks (MFNs) 2) Graph Neural Networks (GNNs).

In the first method, graph refinement is posed as an approximate Bayesian inference task solved using mean-field approximation (MFA) [Jaakkola and Jordan, 1998, Wainwright et al., 2008]. The posterior density over the different subgraphs is approximated with a simpler distribution and the inference is carried out using MFA. We introduce parameterised node and pairwise potentials that capture behaviour of the optimal subgraph corresponding to the underlying tree structure and obtain the MFA update equations within the variational inference framework [Beal et al., 2003]. By unrolling thus obtained MFA update iterations as layers in a feedforward network, we demonstrate the use of gradient descent to learn parameters of this model and point it out to be Mean-Field Network as was suggested in [Li and Zemel, 2014]. We extend the previously published conference work in [Selvan et al., 2018c] in this paper by performing more comprehensive experiments and presenting a thorough comparison with GNNs.

In the second method graph refinement is performed using Graph Neural

Networks. GNNs are a class of recurrent neural networks operating directly on graph structured data [Scarselli et al., 2009, Li et al., 2015] and are now seen as an important step in generalising deep learning models to non-Euclidean domains [Bronstein et al., 2017]. The graph refinement task itself is solved in a supervised setting using a GNN-based encoder that learns edge embeddings based on the input graph, followed by a decoder to obtain probability of edge connections. The idea of using GNNs for graph refinement was initially proposed in [Selvan et al., 2018a], where a GNN-based encoder was used to learn node embeddings and a pairwise decoder was used to predict the edge probabilities. Using node embeddings to predict edge probabilities proved to be inadequate, which we have now addressed in this work by predicting edge probabilities from the learnt edge embeddings.

In addition to proposing MFNs and GNNs as two methods to solve graph refinement tasks, we also study connections between them. In case of the MFN model, the node and pairwise potentials are hand-crafted, incorporating useful prior knowledge. With only a handful of parameters the MFN model requires little supervision and can be seen as an intermediate between a model-based solution and a fully end-to-end training model. On the other hand, the GNN models can be seen as generalisation of message passing algorithms used for inference in probabilistic graphical models [Wainwright et al., 2008] such as MFNs. When used in a supervised setting, as we do, the GNN model can be used to learn task-specific messages to be transacted between nodes of the graph.

We demonstrate usefulness of the graph refinement model and the proposed methods for segmenting tree-structures with an application to extracting airway trees from CT data. We present a three step preprocessing pipeline, following the work in [Selvan et al., 2017], to obtain graph representations of the volumetric image. We perform experiments on the Danish lung cancer screening trial (DLCST) data [Pedersen et al., 2009], comparing the MFN and GNN models to a baseline method that is similar to [Lo et al., 2010]. Based on these experiments we show qualitatively and quantitatively that the proposed methods significantly improve upon the baseline method.

## 4.2 Methods

In this section, we describe the task of graph refinement along with the underlying model assumptions. Based on this model, we present two approaches to performing graph refinement using MFNs and GNNs.

### 4.2.1 Graph Refinement Model

Consider an over-complete, undirected, input graph,  $\mathcal{G}_{\text{in}} : \{\mathcal{V}, \mathcal{E}_{\text{in}}\}$ , with nodes  $i \in \mathcal{V} : |\mathcal{V}| = N$  associated with  $F$ -dimensional features,  $\mathbf{x}_i \in \mathbb{R}^{F \times 1}$  collected into the node feature matrix,  $\mathbf{X} \in \mathbb{R}^{F \times N}$ , and pairwise edges,  $(i, j) \in \mathcal{E}_{\text{in}}$ , described by the input adjacency matrix,  $\mathbf{A}_{\text{in}} \in \{0, 1\}^{N \times N}$ . The goal of graph refinement is to recover a subgraph,  $\mathcal{G}$ , with a subset of edges,  $\mathcal{E} \subset \mathcal{E}_{\text{in}}$ , described by the symmetric output adjacency matrix,  $\mathbf{A} \in \{0, 1\}^{N \times N}$ . This subgraph corresponds to the structure of interest, like airway trees from chest data as studied in this work. We then seek a model,  $f(\cdot)$ , that can recover the subgraph from the input graph,  $f : \mathcal{G}_{\text{in}} \rightarrow \mathcal{G}$ .

### 4.2.2 Mean-Field Networks

We next propose a probabilistic graph refinement model by introducing a random variable that captures the connectivity between any pair of nodes  $i$  and  $j$ :  $s_{ij} \in \{0, 1\}$ , with the probability of the corresponding connection given as  $\alpha_{ij} \in [0, 1]$ . For each node  $i$ , the binary random variables associated with its incident connections are collected into a node connectivity variable  $\mathbf{s}_i = \{s_{ij} : j = 1 \dots N\}$ . At the graph level, all node connectivity variables are collected into a global connectivity variable,  $\mathbf{S} = [\mathbf{s}_1 \dots \mathbf{s}_N]$ .

The graph refinement model is described by the conditional distribution,  $p(\mathbf{S} | \mathbf{X}, \mathbf{A}_{\text{in}})$ , where the node features,  $\mathbf{X}$ , and input adjacency,  $\mathbf{A}_{\text{in}}$ , are observed from the data. This model bears similarities with the hidden MRF models that have been previously used for image segmentation, where the joint distribution is approximated with unary and pairwise energy functions [Zhang et al., 2001, Orlando and Blaschko, 2014].

We use the notion of node potential,  $\phi_i(\mathbf{s}_i)$ , and pairwise edge potential,  $\phi_{ij}(\mathbf{s}_i, \mathbf{s}_j)$ , to express the joint distribution  $p(\mathbf{S}, \mathbf{X}, \mathbf{A}_{\text{in}})$  and relate it to the conditional distribution as

$$\begin{aligned} \ln p(\mathbf{S} | \mathbf{X}, \mathbf{A}_{\text{in}}) &\propto \ln p(\mathbf{S}, \mathbf{X}, \mathbf{A}_{\text{in}}) \\ &= \sum_{i \in \mathcal{V}} \phi_i(\mathbf{s}_i) + \sum_{(i,j) \in \mathcal{E}_{\text{in}}} \phi_{ij}(\mathbf{s}_i, \mathbf{s}_j) - \ln Z \end{aligned} \quad (4.1)$$

where  $\ln Z$  is the normalisation constant. For ease of notation, explicit dependence on observed data in these potentials is not shown. To design suitable potentials for graph refinement we model terms that contribute positively when the nodes or the pairwise connections are likely to belong to the subgraph, and less positively or even negatively otherwise. First, we propose a node potential that captures the importance of a given node in the underlying subgraph,  $\mathcal{G}$ .

For each node  $i \in \mathcal{V}$ , it is given as

$$\phi_i(\mathbf{s}_i) = \sum_{v=0}^D \beta_v \mathbb{I}\left[\sum_j s_{ij} = v\right] + \mathbf{a}^T \mathbf{x}_i \sum_j s_{ij}, \quad (4.2)$$

where  $\sum_j s_{ij}$  is the degree of node  $i$  and  $\mathbb{I}[\cdot]$  is the indicator function. The parameters  $\beta_v \in \mathbb{R}$ ,  $\forall v = [0, \dots, D]$ , can be seen as the prior on the degree per node. We explicitly model and learn this term for up to 2 edges per node and assume uniform prior for  $D > 2$ . Nodes with  $D = 0$  correspond to nodes that do not belong to the subgraph,  $D = 1$  are root or terminal nodes,  $D = 2$  are the most common nodes in the subgraph which are connected to a parent node and a child node. For these cases we explicitly learn the prior weight parameter  $\beta_v \forall v = [0, 1, 2]$ . Further, the individual node features,  $\mathbf{x}_i$ , are combined with  $\mathbf{a} \in \mathbb{R}^{F \times 1}$  to capture a weighted combination of node features that is representative of the node's significance. Contribution of the weighted combination of features of each node to the unary potential is proportional to a node's degree, as seen in the second term in Equation (4.2). That is, a node with more connections is more important to the subgraph and it contributes more to the node potential.

Secondly, the form of pairwise potential should be such that it captures the relation between pairs of nodes and reflects their affinity to be connected. We propose a potential that enforces symmetry in connections, and has terms based on pairwise features for each pair of nodes, given as,

$$\begin{aligned} \phi_{ij}(\mathbf{s}_i, \mathbf{s}_j) = & \lambda(1 - 2|s_{ij} - s_{ji}|) \\ & + (2s_{ij}s_{ji} - 1) \left[ \boldsymbol{\eta}^T |\mathbf{x}_i - \mathbf{x}_j| + \boldsymbol{\nu}^T (\mathbf{x}_i \circ \mathbf{x}_j) \right]. \end{aligned} \quad (4.3)$$

The first term in Equation (4.3) ensures symmetry in connections between nodes, i.e, for nodes  $i, j$  it encourages  $s_{ij} = s_{ji}$ . The parameter  $\boldsymbol{\eta} \in \mathbb{R}^{F \times 1}$  is a weighting of the absolute difference between nodes for each feature dimension. The element-wise feature product term  $(\mathbf{x}_i \circ \mathbf{x}_j)$  with  $\boldsymbol{\nu} \in \mathbb{R}^{F \times 1}$  is a weighted, non-stationary kernel that computes the dot product of node features in a weighted feature space. The second term in Equation (4.3) is multiplied with  $(2s_{ij}s_{ji} - 1)$  to ensure that the contribution to the pairwise potential is positive when both nodes  $i$  and  $j$  are connected to each other, otherwise the contribution is negative.

Returning to the posterior distribution, we note that except for in trivial cases, it is intractable to estimate  $p(\mathbf{S}|\mathbf{X}, \mathbf{A}_{\text{in}})$  in Equation (4.1) and we must resort to approximating it. We take up the variational mean field approximation (MFA) [Jaakkola and Jordan, 1998], which is a structured approach to approximating  $p(\mathbf{S}|\mathbf{X}, \mathbf{A}_{\text{in}})$  with candidates from a class of simpler



distributions:  $q(\mathbf{S}) \in \mathcal{Q}$ . This approximation is performed by iteratively minimizing the exclusive Kullback-Leibler divergence [Jaakkola and Jordan, 1998], or equivalently maximising the evidence lower bound (ELBO) or the variational free energy, given as

$$\mathcal{F}(q_{\mathbf{S}}) = \ln Z + \mathbb{E}_{q_{\mathbf{S}}} \left[ \ln p(\mathbf{S} | \mathbf{X}, \mathbf{A}_{\text{in}}) - \ln q(\mathbf{S}) \right], \quad (4.4)$$

where  $\mathbb{E}_{q_{\mathbf{S}}}$  is the expectation with respect to the distribution  $q_{\mathbf{S}}$ . In MFA, the class of distributions,  $\mathcal{Q}$ , are constrained such that  $q(\mathbf{S})$  can be factored further. In our model, we assume that the existence of each edge is independent of the others, which is enforced by the following factorisation:

$$q(\mathbf{S}) = \prod_{i=1}^N \prod_{j=1}^N q_{ij}(s_{ij}), \quad (4.5)$$

$$\text{where, } q_{ij}(s_{ij}) = \begin{cases} \alpha_{ij} & \text{if } s_{ij} = 1 \\ (1 - \alpha_{ij}) & \text{if } s_{ij} = 0 \end{cases}, \quad (4.6)$$

with  $\alpha_{ij}$  equal to the probability of an edge existing between nodes  $i$  and  $j$ .

Using the potentials from (4.2) and (4.3) in (4.4) and taking expectation with respect to  $q_{\mathbf{S}}$ , we obtain the ELBO in terms of  $\alpha_{ij} \forall i, j = [1, \dots, N]$ . By differentiating this ELBO with respect to any individual  $\alpha_{kl}$ , as elaborated in Appendix 4.4.6, we obtain the following update equation for performing MFA iterations. At iteration  $(t + 1)$ , for each node  $k$ ,

$$\alpha_{kl}^{(t+1)} = \sigma(\gamma_{kl}) = \frac{1}{1 + \exp^{-\gamma_{kl}}} \quad l \in \mathcal{N}_k, \quad (4.7)$$

where  $\sigma(\cdot)$  is the sigmoid activation function,  $\mathcal{N}_k$  are the  $L$  nearest neighbours of node  $k$  based on positional Euclidean distance, and

$$\begin{aligned} \gamma_{kl} = & \prod_{j \in \mathcal{N}_k \setminus l} (1 - \alpha_{kj}^{(t)}) \left\{ \sum_{m \in \mathcal{N}_k \setminus l} \frac{\alpha_{km}^{(t)}}{(1 - \alpha_{km}^{(t)})} [(\beta_2 - \beta_1) \right. \\ & \left. - \beta_2 \sum_{n \in \mathcal{N}_k \setminus l, m} \frac{\alpha_{kn}^{(t)}}{(1 - \alpha_{kn}^{(t)})}] + (\beta_1 - \beta_0) \right\} + \mathbf{a}^T \mathbf{x}_k \\ & + (4\alpha_{lk}^{(t)} - 2)\lambda + 2\alpha_{lk}^{(t)} (\boldsymbol{\eta}^T |\mathbf{x}_k - \mathbf{x}_l| + \boldsymbol{\nu}^T (\mathbf{x}_k \circ \mathbf{x}_l)). \end{aligned} \quad (4.8)$$

After each iteration  $(t)$ , the MFA procedure outputs predictions for the global connectivity variable,  $\boldsymbol{\alpha}^{(t)}$ , with entries  $\alpha_{kl}^{(t)}$  given in Equation (4.7). These MFA iterations are performed until convergence; a reasonable stopping criterion

is when the increase in ELBO is below a small threshold between successive iterations.

The MFA update procedure described in Equation (4.7) and Equation (4.8) resemble the computations in a feed-forward neural network. The predictions from iteration  $(t)$ ,  $\boldsymbol{\alpha}^{(t)}$ , are combined and passed through a non-linear activation function, a sigmoid in our case, to obtain predictions at iteration  $(t+1)$ ,  $\boldsymbol{\alpha}^{(t+1)}$ .  $T$  iterations of MFA can be done with a  $T$ -layered network based on the underlying graphical model. This can be seen as the mean field network (MFN) [Li and Zemel, 2014]. The parameters of our model,  $[\lambda, \beta, \mathbf{a}, \eta, \nu]$ , form weights of such a network and are shared across all layers. Given this setting, parameters for the MFN can be learned by back-propagating any suitable loss,  $\mathcal{L}(\boldsymbol{\alpha}, \mathbf{A}_r)$ , computed between the predicted global connectivity variable at the final iteration  $\boldsymbol{\alpha} = \boldsymbol{\alpha}^{(T)}$  and the reference adjacency,  $\mathbf{A}_r$ . We recover a symmetric adjacency matrix from the predicted global connectivity variable as  $\mathbf{A} = \mathbb{I}[(\boldsymbol{\alpha} > 0.5) \wedge (\boldsymbol{\alpha}^T > 0.5)]$ , because symmetry is not enforced on the predicted global connectivity variable i.e, the equality  $\alpha_{ij} = \alpha_{ji}$  does not always hold. This is because of the MFA factorisation in Equation (4.5) where we assume edges to be independent of each other. Details of the MFN training procedure are presented in Section 4.2.4.

### 4.2.3 Graph Neural Networks

With MFN in Section 4.2.2, we presented a hand-crafted model to perform graph refinement in an unsupervised setting. In this section, we investigate if the messages transacted between nodes according to Equations (4.7) and (4.8) in the MFN can be learnt in a supervised setting using Graph Neural Networks.

GNNs are neural networks that operate directly on graph structured data by passing local messages between nodes [Scarselli et al., 2009, Li et al., 2015]. Several closely related formulations of GNNs are prevalent in literature [Kipf and Welling, 2017, Bronstein et al., 2017, Gilmer et al., 2017, Kipf et al., 2018]. In this work, we focus on a model variant, termed a graph auto-encoder (GAE), first introduced in [Kipf and Welling, 2016]. A GAE is comprised of an encoder-decoder pair that is jointly trained to learn expressive node and/or edge embeddings of the input graph. We extend the preliminary work in [Selvan et al., 2018a], where graph refinement was performed using only node embeddings, and a pairwise decoder was used to predict edge probabilities. A GNN model that only uses node embeddings is referred to as the node-GNN [Li et al., 2015, Kipf and Welling, 2017, Yoon et al., 2018]. In comparison, the current model, uses an encoder to learn edge embeddings and then a decoder that predicts edge probability from these edge embeddings. These GNN models are referred to as the edge-GNN [Gilmer et al., 2017, Yoon et al., 2018, Kipf et al., 2018].

The graph refinement task, as formulated in Section 4.2.1, provides a conducive setting to use GNN based encoder-decoder pairs to estimate the model,  $f : \mathcal{G}_{\text{in}} \rightarrow \mathcal{G}$ . The GNN model, in our case, is used in a supervised setting to learn edge embeddings from which the subgraphs of interest can be reconstructed. Joint training of encoder-decoder pair yields an encoder that maps the input node features,  $\mathbf{X}$ , to node embeddings, computes the corresponding edges based on the input adjacency matrix,  $\mathbf{A}_{\text{in}}$ , and obtains edge embeddings. The decoder, then, uses the learnt edge embeddings to predict the global connectivity variable,  $\alpha$ .

Following the notation in [Kipf et al., 2018], we present a GNN based encoder with a receptive field of two, that is with two GNNs (identified by the superscripts):

$$\text{Node Embedding:} \quad \mathbf{h}_j^{(1)} = g_n(\mathbf{x}_j) \quad (4.9)$$

$$\text{Node-to-Edge mapping:} \quad \mathbf{h}_{(i,j)}^{(1)} = g_{n2e}([\mathbf{h}_i^{(1)}, \mathbf{h}_j^{(1)}]) \quad (4.10)$$

$$\text{Edge-to-Node mapping:} \quad \mathbf{h}_j^{(2)} = g_{e2n}\left(\sum_i^{N_j} \mathbf{h}_{(i,j)}^{(1)}\right) \quad (4.11)$$

$$\text{Node-to-Edge mapping:} \quad \mathbf{h}_{(i,j)}^{(2)} = g_{n2e}([\mathbf{h}_i^{(2)}, \mathbf{h}_j^{(2)}]) \quad (4.12)$$

where, each of the  $g_{\dots}(\cdot)$  above is a 2-layered multi-layer perceptron (MLP) with rectified linear unit activations, dropout between the two hidden layers, skip connections and layer normalisation. Equation (4.9) describes the node embedding corresponding to the first GNN,  $\mathbf{h}_j^{(1)}$ . The MLP,  $g_n(\cdot)$ , has  $F$  input channels and  $E$  output channels transforming the  $F$ -dimensional input node features into  $E$ -dimensional node embedding. Edge embedding,  $\mathbf{h}_{(i,j)}^{(1)}$  for a pair of nodes,  $(i, j)$  is obtained by simply concatenating the corresponding node features and propagating these features through the edge MLP, as described in Equation (4.10). The edge MLPs,  $g_{n2e}(\cdot)$  have  $2E$  input channels and  $E$  output channels. Going from these edge embeddings to node representation is performed by simply summing over all the incoming edges to any given node  $j$  from its neighbourhood,  $\mathcal{N}_j$  according to Equation (4.11). A new node embedding is obtained by propagating these node features through the second node MLP,  $g_{e2n}(\cdot)$  with  $E$  input and output channels, as described in Equation (4.11). The second edge MLP,  $g_{n2e}(\cdot)$  also has  $2E$  input and  $E$  output channels. Finally, the output from the encoder,  $\mathbf{h}_{(i,j)}^{(2)}$ , in Equation (4.12) is the  $E$ -dimensional edge embedding used to predict the edge probabilities with a simple decoder.

The decoder is given as:

$$\alpha_{ij} = \sigma(g_{dec}(\mathbf{h}_{(i,j)}^{(2)})) \quad (4.13)$$

where,  $g_{dec}$  is a 1D convolutional layer with one output unit and  $\sigma(\cdot)$  is the sigmoid activation function. This decoding operation converts the  $E$ -dimensional edge embedding into a single scalar for each edge and the sigmoid function yields the edge probability,  $\alpha_{ij}$ . These are entries for predicted global connectivity variable,  $\alpha$ , similar to the predictions using MFN in Equation (4.7). As with MFN, the GNN model loss is computed based on edge probability predictions,  $\alpha$ , and the reference adjacency matrices,  $\mathcal{L}(\alpha, \mathbf{A}_r)$ .

Although the GNN model description above is for individual nodes and edges, these can be vectorised for faster implementation [Kipf et al., 2018]. Also, the receptive field of the encoder can be easily increased by stacking more GNNs i.e, successive repetition of pairs of Equations (4.11) and (4.12).

#### 4.2.4 Loss Function

Both, MFN and GNN, models output predictions for the global connectivity variable,  $\alpha$ , which has entries corresponding to the probability of existence of each edge. From a loss point of view, this is similar to the binary classification, as the reference adjacency matrix,  $\mathbf{A}_r$ , has binary entries indicating the presence of edges in the underlying subgraph. In most applications, the graphs are sparse as the “edge” class is in minority. To overcome challenges in training due to such class skew we use the Dice loss [Milletari et al., 2016] for optimising both the models, for its inherent ability to account for the class imbalance. Dice loss is given as:

$$\mathcal{L}(\alpha, \mathbf{A}_r) = 1 - \frac{2 \sum_{i,j=1}^N \alpha_{ij} A_{ij}}{\sum_{i,j=1}^N \alpha_{ij}^2 + \sum_{i,j=1}^N A_{ij}^2}, \quad (4.14)$$

where,  $A_{ij}$  are the individual binary entries in the reference adjacency matrix.

## 4.3 Experiments and Results

### 4.3.1 Airway Tree Extraction as Graph Refinement

Both the MFN and GNN models presented are general models that can be adapted and applied to any graph refinement task. Here we present extraction of airway tree centerlines from CT images as a graph refinement task, and describe the specific features used for this application.

#### Preprocessing

The image data is pre-processed to convert it into a graph-like format. First, the 3D CT image data is converted into a probability map using a trained voxel

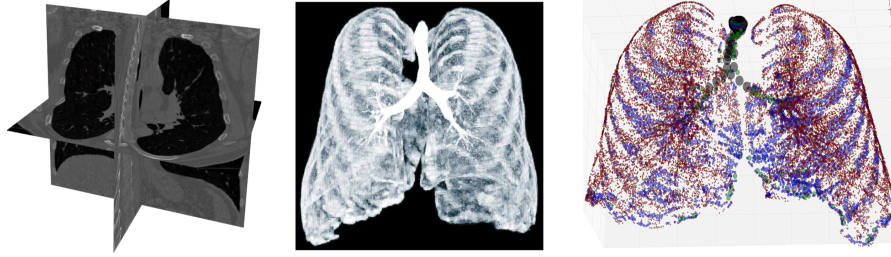


Figure 4.1: The pre-processing to transform the input image (left) into a probability image (center) and then into graph format (right). Nodes in the graph are shown in scale (as different colours) to capture the variations in their local radius.

classifier according to Lo et al. (2010) [Lo et al., 2010]. This step converts intensity per voxel into a probability of that voxel belonging to the airway lumen. These probability images are transformed to sparse representation using a simple multi-scale blob detector. Next, we obtain perform Bayesian smoothing, with process and measurement models that model individual branches in an airway tree, using the method of [Selvan et al., 2017]. This three-step pre-processing procedure yields a graph output of the input image data, as illustrated in Figure 4.1. Each node in this graph is associated with a 7-dimensional Gaussian density comprising of spatial location  $(x, y, z)$  in the image, local radius  $(r)$ , and orientation  $(v_x, v_y, v_z)$ , such that  $\mathbf{x}_i = [\mathbf{x}_\mu^i, \mathbf{x}_{\sigma_2}^i]$ , comprising mean,  $\mathbf{x}_\mu^i \in \mathbb{R}^{7 \times 1}$ , and variance for each feature,  $\mathbf{x}_{\sigma_2}^i \in \mathbb{R}^{7 \times 1}$ . To obtain an initial connectivity,  $\mathbf{A}_{in}$ , we connect nodes to their 10 nearest neighbours. Thus obtained graphs, with node features  $\mathbf{x}_i \in \mathbb{R}^{14 \times 1}$ , are the input graphs for both models. Figure 4.2 (left) visualises an over-connected input graph.

### Adapting the MFN model

The node and pairwise potentials in equations (4.2) and (4.3) are general and applicable to commonly encountered trees. Due to the nature of features extracted for the nodes in Section 4.3.1, one of the terms in the pairwise potential in Equation (4.3) requires a minor modification. The factor in Equation (4.3) associated with  $\boldsymbol{\eta}$  is the absolute difference in node features,  $|\mathbf{x}_i - \mathbf{x}_j|$ . The first three dimensions of the node features are the spatial location  $\mathbf{x}_p = [x, y, z]$ ; we normalise the absolute distance between the two nodes with the average radius of the two nodes, i.e.,  $|\mathbf{x}_p^i - \mathbf{x}_p^j| / (r^i + r^j)$ , as the relative positions of nodes are proportional to their scales in the image.

### Reference Adjacency Matrices

Reference adjacency matrices are obtained from the reference segmentations using Bayesian smoothing to extract individual airway branches, from probability images obtained using voxel classification of CT data [Lo et al., 2010], closely following the method in [Selvan et al., 2017]. Thus extracted branches that are inside the corresponding reference segmentations are connected using a spanning tree algorithm to obtain a single connected tree, yielding reference adjacency matrices used for training both the GNN and MFN models. A sample input graph along with the connections based on the reference adjacency matrix is shown in Figure 4.2 (center) .

### Obtaining binary segmentations

The output of graph refinement models yields connectivity information about the airway centerlines. For evaluation purposes, we convert the predicted subgraph into a binary segmentation. This is done by drawing binary voxels within a tubular region that interpolates the scales of the two nodes, along edges given by  $\mathbf{A} = \mathbb{I}[(\boldsymbol{\alpha} > 0.5) \wedge (\boldsymbol{\alpha}^T > 0.5)]$ . One such binary segmentation is visualised in Figure 4.2 (right).

#### 4.3.2 Data

The experiments were performed on 3-D, low-dose CT, chest scans from the Danish lung cancer screening trial [Pedersen et al., 2009]. All scans have voxel resolution of approximately  $0.78 \times 0.78 \times 1 \text{ mm}^3$ . We use two non-overlapping sets of 32 scans and 100 scans for training and evaluation purposes. The 32 scans in the first subset have reference segmentations that are treated as the ground truth for the purpose of evaluations, referred to as the *reference dataset*. These reference segmentations are obtained by combining results from two previous airway segmentation methods [Lo et al., 2009, Lo et al., 2010] that are corrected by an expert user. First of these methods uses a trained voxel classifier to distinguish airway regions from the background to yield probability images, and airway trees are extracted with region growing on these probabilities using an additional vessel similarity measure [Lo et al., 2010]. The second method extracts airways by extending locally optimal paths on the same probability images [Lo et al., 2009]. The second set comprising 100 scans has automatic segmentations obtained using [Lo et al., 2009]. As the reference dataset is relatively small, we use the second set of 100 scans to perform pre-training and to tune hyperparameters of both the models, referred to as the *pre-training dataset*.

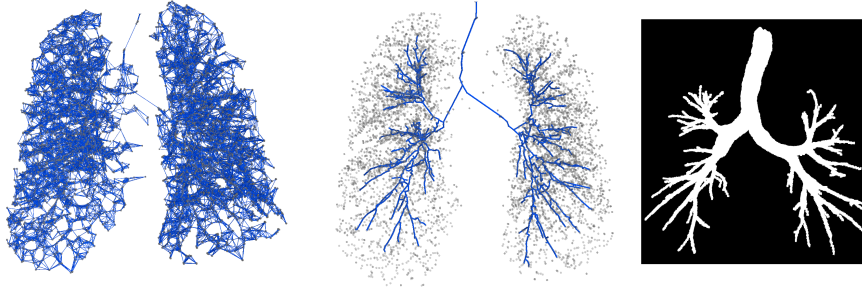


Figure 4.2: Input graph derived from a chest scan depicting the initial connectivity based on  $\mathbf{A}_{in}$  between nodes (left). Nodes of input graph (grey dots) overlaid with connections derived from the reference adjacency matrix,  $\mathbf{A}_r$  (center). Binary volume segmentation obtained from the reference adjacency matrix and corresponding node features (right).

### 4.3.3 Error Measure

Comparison of the graph refinement performance of the MFN and GNN models is done based on computing Dice similarity coefficient

$$Dice = \frac{2|\mathbf{A} * \mathbf{A}_r|}{|\mathbf{A}| + |\mathbf{A}_r|}, \quad (4.15)$$

where the product in the numerator is element-wise.

To evaluate the binary segmentations obtained using the procedure in Section 4.3.1, centerline distance is used. Centerlines are extracted from the binary segmentations using a 3-D thinning algorithm [Homann, 2007] to be consistent in the evaluation of all comparing methods. The extracted centerlines are compared with the corresponding reference centerlines using an error measure that captures the average centerline distance. It is defined as:

$$d_{err} = \frac{d_{FP} + d_{FN}}{2}, \quad (4.16)$$

where  $d_{FP}$  captures the errors due to false positive branches – it is the average minimum Euclidean distance from segmented centerline points to reference centerline points – and  $d_{FN}$  captures the errors due to false negatives – it is the average minimum Euclidean distance from reference centerline points to segmentation centerline points.

### 4.3.4 Training the models

Training of both MFN and GNN models was performed in three stages: hyperparameter tuning, pre-training and final model training, using the Dice loss in Equation (4.14). Hyperparameters such as the number of layers, training iterations and learning rate were tuned, and pre-training of both models was performed, using the *pre-training* dataset. The model parameters themselves were trained and evaluated using an 8-fold cross validation procedure using the 32 scans in the *reference* dataset. All experiments were performed using a GPU with 12GB memory and the code was implemented in PyTorch. The AMSGrad variant of Adam optimizer was used for optimization [Reddi et al., 2018] with an initial learning rate of 0.005.

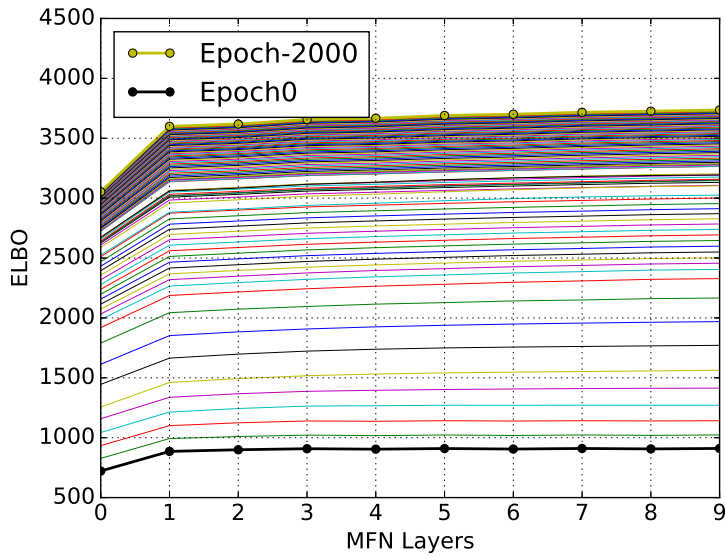


Figure 4.3: Plot showing the evolution of ELBO with each epoch and across each MFN layer for the MFN model. A clear trend of increase in ELBO within each epoch and across epochs is seen.

#### MFN parameters

The most important hyperparameter in the MFN model is the number of layers  $T$ , equivalently the number of MFA iterations. Based on our initial experiments of observing the evolution of ELBO (see Figure 4.3) we set  $T = 10$ . The number of training epochs was set to 2000. Each input graph, on average, has 8000



Table 4.1: Performance comparison of four methods: Voxel classifier (Vox.), Bayesian smoothing (Bayes.), MFN and GNN models, using centerline distance based on 8-fold cross validation.

Method	Dice(%)	$d_{FP}$ (mm)	$d_{FN}$ (mm)	$d_{err}$ (mm)
Vox.	–	$3.624 \pm 0.776$	$5.155 \pm 0.580$	$4.389 \pm 0.441$
Bayes.	–	$3.921 \pm 0.612$	$4.218 \pm 0.334$	$4.069 \pm 0.476$
MFN	<b><math>86.5 \pm 2.5</math></b>	$3.599 \pm 0.583$	$3.491 \pm 0.295$	$3.595 \pm 0.321$
GNN	$84.8 \pm 3.3$	<b><math>3.045 \pm 0.329</math></b>	<b><math>2.951 \pm 0.757</math></b>	<b><math>2.998 \pm 0.399</math></b>

nodes. We create sub-images comprising 500 nodes from each graph to input to the MFN to reduce memory utilisation. Batch size of 12 images (comprising all sub-images corresponding to an input graph) was used in the training procedure.

### GNN model parameters

Based on the *pre-training* dataset, we designed an architecture for the GNN model comprising an encoder with a receptive field of 2 as described in Section 4.2.3. Each of the MLPs,  $g_{\dots}(\cdot)$ , used in the encoder in Equations (4.9)–(4.12) has two hidden layers and the number of channels per layer parameter  $E = 8$ . A dropout rate of 0.5 was used between each layer in the MLPs. Number of training epochs for the GNN model was set to 500. Batch size of 12 was used during training.

### 4.3.5 Results

We compare performance of the MFN and GNN models to each other, and with a baseline airway extraction method that uses region growing on probability images obtained using a voxel classifier. The baseline method is close to the method in [Lo et al., 2010]. Further, as the input to the graph refinement methods were based on Bayesian smoothing in [Selvan et al., 2017], we also report the results for Bayesian smoothing method and it is discussed in Section 4.4.1.

Dice accuracy, in Equation (4.15), for the two graph-refinement models is reported in the second column in Table 4.1. Remaining columns in Table 4.1 report the centerline distance error based performance. A standard box plot visualisation of the two component distance measures is presented in Figure 4.4 for each of the methods. Test set centerline predictions for two cases along with the reference segmentations for all the methods are visualised in Figure 4.5.

Based on the centerline distance measure reported in Table 4.1 we see that both the MFN and GNN models show significant overall improvement

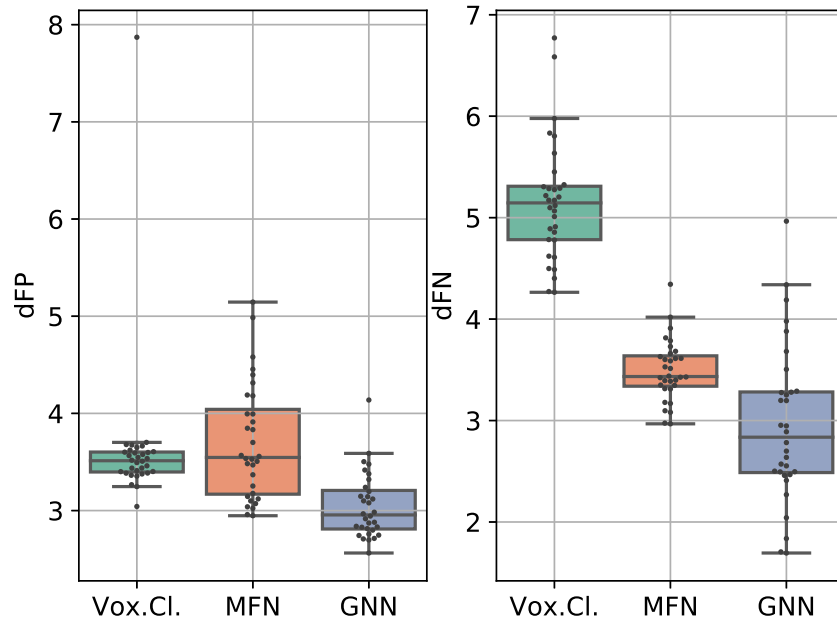


Figure 4.4: Standard box plots showing false positive error,  $d_{FP}$  (mm), and false negative error,  $d_{FN}$  (mm), for all 32 images in the reference set, for the three methods.

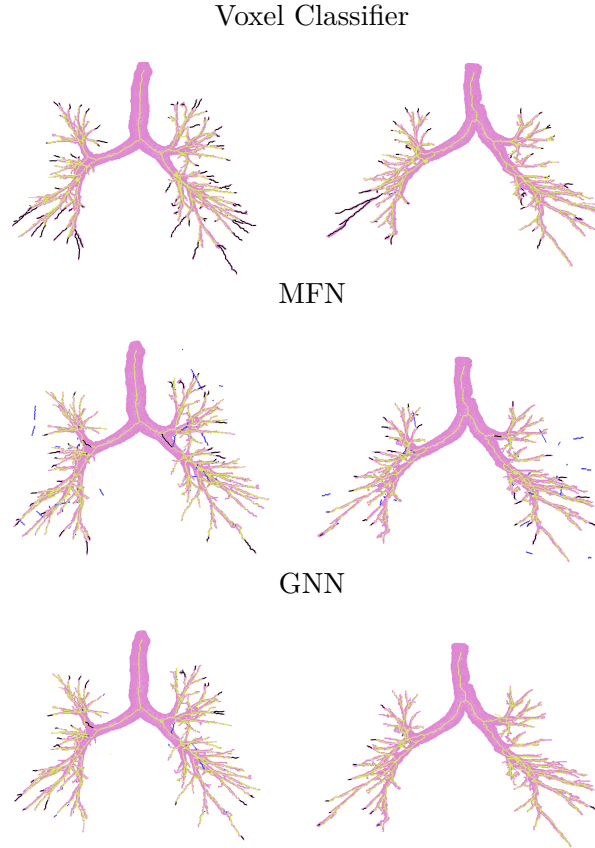


Figure 4.5: Predicted centerlines for two test cases (along each column) from the three methods overlaid with the reference segmentation (pink surface). In each case different colours are used to show true positive (yellow), false positive (blue) and false negative (black) branches.

captured in  $d_{err}$  ( $p < 0.001$ ) when compared to the voxel classifier. Both methods specifically show large and significant improvement in  $d_{FN}$  ( $p < 0.001$ ), indicating their capability to detect more branches than the voxel classifier, which is evident when comparing the entries for  $d_{FN}$  in Figure 4.5 (right). The improvements on  $d_{FP}$  when compared to the voxel classifier is not significant for the MFN model ( $p = 0.668$ ), whereas for the GNN model it is significant ( $p < 0.001$ ).

When comparing the performance between the MFN and GNN models in Table 4.1, we see a significant improvement using the GNN model in  $d_{err}$  ( $p < 0.005$ ). However, the MFN model obtains a higher Dice accuracy when

compared to the GNN model as reported in the second column of Table 4.1. The reported significance values in the following discussions are based on paired sample  $t$ -tests.

## 4.4 Discussion and Conclusions

In this work, we formulated extraction of tree-like structures from volumetric data, with a focus on airways, as a graph refinement task. Detecting small branches and overcoming occlusions during extraction of airways from CT data is challenging. By posing tree extraction as a graph refinement task we presented an exploratory approach that, to a large extent, overcomes these challenges. Some existing airway segmentation methods also have taken up an exploratory approach, like the work in [Bauer et al., 2015], where candidate airway branches are obtained using a tube detection filter and tree reconstruction is performed as a two-step graph based optimisation. Candidate airway branches form nodes of this graph and plausible edges between these nodes are worked out in the first step of the optimisation. In the second step of the optimisation, sub-trees below a certain score are pruned away. In comparison to [Bauer et al., 2015], the proposed graph refinement setting operates on nodes that are local regions of interest, and reconstructs branches and connections between branches simultaneously from these nodes. This graph refinement framework can be seen as taking up a more global approach to tree reconstruction, as it does not rely on thresholding local sub-trees.

Two models for graph refinement based on mean-field networks and graph neural networks were presented, evaluated on chest CT data and compared to a baseline method that is similar to [Lo et al., 2010]. The method in [Lo et al., 2010] was one of the top performing methods in EXACT’09 airway extraction challenge [Lo et al., 2012] and forms a useful baseline for comparison.

### 4.4.1 Influence of Preprocessing

The input to the two graph refinement models is based on a preprocessing step that is based on Bayesian smoothing in [Selvan et al., 2017] as described in Section 4.3.1. To isolate the improvements due to the graph refinement models, we report the centerline error for the predictions from the Bayesian smoother in Table 4.1. Following the reporting of Bayesian smoothing results in [Selvan et al., 2018c], the centerline error measure is reported for the Bayesian smoothing method combined with the voxel classifier. This is because the output of the Bayesian smoother is obtained by thresholding the average total variance of each branch, resulting in a collection of possible airway branches as the

prediction [Selvan et al., 2017]. From the  $d_{err}$  entries in Table 4.1 we notice that both the graph refinement models show large and significant improvement ( $p < 0.001$ ) when compared to the combined predictions of Bayesian smoothing method and the voxel classifier reported in the second row. Based on this observation we claim that the large portion of performance improvements are due to the graph refinement procedures.

#### 4.4.2 MFN model

The main contribution within the presented MFN framework is our formulation of unary and pairwise potentials in (4.2) and (4.3). By designing such potentials to reflect the nature of tasks we are interested in, the MFN model can be applied to diverse applications. For instance, it has been showed that information from pulmonary vessels can be used to improve airway segmentation in [Lo et al., 2010]. Modeling potential functions that take this information into account and encode the relation between vessel and airway branches can be done with MFN. Also, semantic segmentation tasks that predict on voxel labels can also be modelled in the MFN setting, bearing similarities with the models used in [Orlando and Blaschko, 2014].

The MFN model can be seen as an intermediate between an entirely model-based solution and an end-to-end learning approach. It can be interpreted as a structured neural network where the interactions between layers are based on the underlying graphical model, while the parameters of the model are learned from data. This, we believe, presents an interesting link between probabilistic graphical models and neural network-based learning.

The significant improvement in  $d_{FN}$  of the MFN model when compared to the voxel classifier is an indication of its ability to detect more branches than the voxel classifier. Low improvement in  $d_{FP}$  ( $p = 0.668$ ) can be due to some false positives that are disconnected, spurious branches predicted by the MFN model, as seen in Figure 4.5 (second row). This behaviour can be attributed to the fact there are no explicit tree structure constraints enforced in the potential functions of the MFN model in Equations (4.2) and (4.3).

#### 4.4.3 GNN model

In our earlier work in [Selvan et al., 2018a], we first introduced the GNN model for graph refinement tasks. In that work, however, the GNN model was used to learn node embeddings, using node GNNs. A pairwise decoder was then used to predict edge probabilities from the learnt node embeddings. With our experiments we found out the model was performing inadequately and hypothesised use of more training data and updating the model to using edge

GNNs could be beneficial. With the model presented here, in Section 4.2.3, we introduced edge GNNs in the encoder to explicitly represent the edges, in order to learn edge embeddings. By jointly training the encoder-decoder pair now, we use the learnt edge embeddings to predict the probability of edges.

The graph encoder used in this work consists of two GNN layers meaning, nodes of the GNN have access to messages from first and second order neighbourhoods. The choice of this receptive field was based on initial experiments done using the *pretraining* dataset to finalise the network architecture. This receptive field can be further increased by adding GNN layers. In theory, a sufficiently deep GNN-based encoder should allow each node to receive messages from all other nodes. For the graph refinement task, we observed a receptive field of two to be sufficient.

From Table 4.1 we observe that the GNN model shows significant improvement on both  $d_{FN}$  and  $d_{FP}$  ( $p < 0.001$ ) when compared to the voxel classifier. That is, the GNN model not only detects more branches than the voxel classifier but also predicts fewer false branches as seen in Figure 4.5 (third row).

#### 4.4.4 Comparison between MFN and GNN models

The MFN model update Equations (4.8) and (4.7) reveal the message passing nature of the underlying inference procedure [Wainwright et al., 2008]. The state of each node i.e., the edge update message between nodes  $k$  and  $l$  is dependent on their corresponding data terms and all neighbours of node  $k$  except node  $l$ . These messages transacted in a  $T$ -layered MFN are hand-crafted based on the model in Equations (4.2) and (4.3) and deriving an analytical solution that guarantees increase in ELBO. However, deriving such analytical solutions might not be feasible for all scenarios.

As GNNs can be seen as generalisation of message passing based inference methods [Gilmer et al., 2017, Yoon et al., 2018], with a capability of learning complex task-specific messages, an interesting connection with the MFN model can be made. Given sufficient training data, in principle, the GNN model should be able to learn messages to approximate the same posterior density as the MFN model. In the context of graph refinement, based on the performances of the two models we observe that, perhaps, the GNN model is able to learn more expressive messages than the MFN model.

The centerline error based performance reported for the MFN and GNN models in Table 4.1, where we see a significant improvement using the GNN model in  $d_{FN}$  ( $p < 0.005$ ), implies that the GNN model is able to detect more missing branches than the MFN model. Also, there is a reduction in  $d_{FP}$  for the GNN model; this is due to several spurious and disconnected branches predicted

by the MFN model. However, the GNN model predicts fewer disconnected edges, indicating that the model is able to learn the behaviour of edges belonging to a connected tree. This is clearly captured in the visualisations in Figure 4.5.

From a graph refinement perspective, we see the MFN model score higher in dice accuracy (second column of Table 4.1). This is contrary to the centerline distance performance but can be explained by noticing that each edge in the dice accuracy in Equation (4.15) has the same importance. That is, edges between nodes in branches of large and small radii have the same importance. However, a missing edge in a branch of large radius can contribute more to the centerline distance than a missing edge in a branch of smaller radius.

The GNN model used here, is relatively more complex, has  $\approx 3150$  tunable weights. The MFN model has a small set of tunable parameters  $[\lambda, \mathbf{a}, \beta, \eta, \nu]$ ; in all 46 scalar weights. Each training epoch containing 24 training images for the MFN model takes about 2s and 1s for the GNN model. The implementation of GNN model takes advantage of sparse matrix operations, for  $O(|\mathcal{E}_{\text{in}}|)$  computational complexity. A similar sparse implementation can further reduce the computation time for the MFN model.

#### 4.4.5 Limitations and Future Work

The preprocessing performed in Section 4.3.1 is one possible way of obtaining graphs from image data as demonstrated in this work. A natural next step is to use more powerful local feature extractors based on CNNs and also have the initial graph extraction as part of the learning set-up. Initial work involving sequential training of feature extraction using CNNs and GNNs for learning global connectivity has been proposed in [Shin et al., 2018] for 2-D vessel segmentation tasks. A joint end-to-end training procedure that dynamically extracts graphs from image data and performs graph refinement can be a challenging, but very interesting direction. These types of models, where the CNNs are used as local feature extractors and GNNs are used on top to model the global connectivity can be useful also to reduce the massive memory footprints of CNN models when used for volume level prediction tasks.

In the MFN model, we currently only use a linear data term in the node potential,  $\mathbf{a}^T \mathbf{x}_i$  in (4.2), and a pairwise potential,  $\nu^T(\mathbf{x}_i \circ \mathbf{x}_j)$  in (4.3). There are possibilities of using more complex data terms to learn more expressive features. For example, use of higher order potentials that take more than two nodes jointly into account can be beneficial in alleviating some of the false positive errors. These errors are revealed as the spurious predictions in Figure 4.5, yielding a subgraph that is not a single, connected, tree.

While the output of the GNN has very few disconnected branches, the output

in all cases is not a fully connected tree. Incorporating tree enforcing constraints, either in the loss function or, in the GNN model could be beneficial.

#### 4.4.6 Conclusion

In this work, we presented a graph refinement setting for the extraction of tree-structures from volumetric data, with a focus on airway extraction. We proposed two novel methods to perform graph refinement based on MFNs and GNNs. Both models were demonstrated to be exploratory methods, which were able to track more branches than the baseline method.

We evaluated the two methods in their ability to extract airway trees from CT data and compared them to a baseline method. With our experiments we have shown that both the MFN and GNN models perform significantly better than the baseline method on the average centerline distance measure. Between the MFN and GNN models, the GNN model is able to detect more branches with fewer false positives. We have also presented connections between the MFN and GNN models. GNNs are more complex models which can be seen as generalisation of MFN models, while the MFN models are simpler and can be viewed as a structured GNNs based on underlying graphical models.

#### Acknowledgements

This work was funded by the Independent Research Fund Denmark (DFR) and SAP SE.

## Appendix

We detail the procedure for obtaining the mean field approximation update equations in (6) and (7) starting from the variational free energy in equation (4). We start by repeating the expression for the node and pairwise potentials.

#### Node potential:

$$\phi_i(\mathbf{s}_i) = \sum_{v=0}^2 \beta_v \mathbb{I}\left[\sum_j s_{ij} = v\right] + \mathbf{a}^T \mathbf{x}_i \sum_j s_{ij}, \quad (4.17)$$

#### Pairwise potential:

$$\begin{aligned} \phi_{ij}(\mathbf{s}_i, \mathbf{s}_j) &= (2s_{ij}s_{ji} - 1) \left[ \boldsymbol{\eta}^T |\mathbf{x}_i - \mathbf{x}_j| + \boldsymbol{\nu}^T (\mathbf{x}_i \circ \mathbf{x}_j) \right] \\ &+ \lambda(1 - 2|s_{ij} - s_{ji}|) \end{aligned} \quad (4.18)$$



The variational free energy is given as,

$$\mathcal{F}(q_{\mathbf{S}}) = \ln Z + \mathbb{E}_{q_{\mathbf{S}}} \left[ \ln p(\mathbf{S} | \mathbf{X}, \mathbf{A}_{\text{in}}) - \ln q(\mathbf{S}) \right]. \quad (4.19)$$

Plugging in (4.17) and (4.18) in (4.19), we obtain the following:

$$\begin{aligned} \mathcal{F}(q_{\mathbf{S}}) = & \ln Z + \mathbb{E}_{q_{\mathbf{S}}} \left[ \sum_{i \in \mathcal{V}} \left\{ \beta_0 \mathbb{I} \left[ \sum_j s_{ij} = 0 \right] \right. \right. \\ & + \beta_1 \mathbb{I} \left[ \sum_j s_{ij} = 1 \right] + \beta_2 \mathbb{I} \left[ \sum_j s_{ij} = 2 \right] + \mathbf{a}^T \mathbf{x}_i \sum_j s_{ij} \left. \right\} \\ & + \sum_{(i,j) \in \mathcal{E}_{\text{in}}} \left\{ \lambda (1 - 2|s_{ij} - s_{ji}|) + (2s_{ij}s_{ji} - 1) \left[ \boldsymbol{\eta}^T |\mathbf{x}_i - \mathbf{x}_j| \right. \right. \\ & \left. \left. + \boldsymbol{\nu}^T (\mathbf{x}_i \circ \mathbf{x}_j) \right] \right\} - \ln q(\mathbf{S}) \right]. \end{aligned} \quad (4.20)$$

We next take expectation  $\mathbb{E}_{q_{\mathbf{S}}}$  using the mean-field factorisation that  $q(\mathbf{S}) = \prod_{i=1}^N \prod_{j \in \mathcal{N}_i} q_{ij}(s_{ij})$  and the fact that  $\Pr\{s_{ij} = 1\} = \alpha_{ij}$  we simplify each of the factors :

$$\begin{aligned} & \mathbb{E}_{q_{\mathbf{S}}} \left[ \beta_0 \mathbb{I} \left[ \sum_j s_{ij} = 0 \right] \right] \\ & = \mathbb{E}_{q_{i1} \dots q_{iN}} \beta_0 \mathbb{I} \left[ \sum_j s_{ij} = 0 \right] = \beta_0 \prod_{j \in \mathcal{N}_i} (1 - \alpha_{ij}). \end{aligned} \quad (4.21)$$

Similarly,

$$\mathbb{E}_{q_{\mathbf{S}}} \left[ \beta_1 \mathbb{I} \left[ \sum_j s_{ij} = 1 \right] \right] = \beta_1 \prod_{j \in \mathcal{N}_i} (1 - \alpha_{ij}) \sum_{j \in \mathcal{N}_i} \frac{\alpha_{im}}{(1 - \alpha_{im})} \quad (4.22)$$

and

$$\begin{aligned} & \mathbb{E}_{q_{\mathbf{S}}} \left[ \beta_2 \mathbb{I} \left[ \sum_j s_{ij} = 2 \right] \right] \\ & = \beta_2 \prod_{j \in \mathcal{N}_i} (1 - \alpha_{ij}) \sum_{m \in \mathcal{N}_i} \sum_{n \in \mathcal{N}_i \setminus m} \frac{\alpha_{im}}{(1 - \alpha_{im})} \frac{\alpha_{in}}{(1 - \alpha_{in})}. \end{aligned} \quad (4.23)$$

Next, we focus on the pairwise symmetry term:

$$\mathbb{E}_{q_{\mathbf{S}}} \left[ \lambda (1 - 2|s_{ij} - s_{ji}|) \right] = \lambda (1 - 2(\alpha_{ij} + \alpha_{ji}) + 4\alpha_{ij}\alpha_{ji}) \quad (4.24)$$

Using these simplified terms, and taking the expectation over the remaining terms, we obtain the ELBO as,

$$\begin{aligned}
\mathcal{F}q_{\mathbf{s}} &= \ln Z + \sum_{i \in \mathcal{V}} \prod_{j \in \mathcal{N}_i} (1 - \alpha_{ij}) \left\{ \beta_0 + \sum_{m \in \mathcal{N}_i} \frac{\alpha_{im}}{(1 - \alpha_{im})} \left[ \beta_1 \right. \right. \\
&+ \left. \beta_2 \sum_{n \in \mathcal{N}_i \setminus m} \frac{\alpha_{in}}{(1 - \alpha_{in})} \right] + \mathbf{a}^T \mathbf{x}_i \sum_j \alpha_{ij} \left. \right\} + \sum_{i \in \mathcal{V}} \sum_{j \in \mathcal{N}_i} \left\{ 4\alpha_{ij}\alpha_{ji} \right. \\
&+ \left. \lambda(1 - 2(\alpha_{ij} + \alpha_{ji})) - (\alpha_{ij} \ln \alpha_{ij} + (1 - \alpha_{ij}) \ln(1 - \alpha_{ij})) \right. \\
&+ \left. (2\alpha_{ij}\alpha_{ji} - 1) \left[ \boldsymbol{\eta}^T |\mathbf{x}_i - \mathbf{x}_j| + \boldsymbol{\nu}^T (\mathbf{x}_i \circ \mathbf{x}_j) \right] \right\}. \tag{4.25}
\end{aligned}$$

We next differentiate ELBO in (4.25) wrt  $\alpha_{kl}$  and set it to zero.

$$\begin{aligned}
\frac{\partial \mathcal{F}q_{\mathbf{s}}}{\partial \alpha_{kl}} &= - \left[ \ln \frac{\alpha_{kl}}{1 - \alpha_{kl}} \right] + \prod_{j \in \mathcal{N}_k \setminus l} (1 - \alpha_{kj}) \left\{ \sum_{m \in \mathcal{N}_k \setminus l} \frac{\alpha_{km}}{(1 - \alpha_{km})} \right. \\
&\left[ (\beta_2 - \beta_1) - \beta_2 \sum_{n \in \mathcal{N}_k \setminus l, m} \frac{\alpha_{kn}}{(1 - \alpha_{kn})} \right] + (\beta_1 - \beta_0) \left. \right\} + \mathbf{a}^T \mathbf{x}_k + \\
&(4\alpha_{lk} - 2)\lambda + 2\alpha_{lk} (\boldsymbol{\eta}^T |\mathbf{x}_k - \mathbf{x}_l| + \boldsymbol{\nu}^T (\mathbf{x}_k \circ \mathbf{x}_l)). \quad = 0 \tag{4.26}
\end{aligned}$$

From this we obtain the MFA update equation for iteration  $(t+1)$  based on the states from  $(t)$ ,

$$\alpha_{kl}^{(t+1)} = \sigma(\gamma_{kl}) = \frac{1}{1 + \exp^{-\gamma_{kl}}} \quad \forall k = \{1 \dots N\}, l \in \mathcal{N}_k \tag{4.27}$$

where  $\sigma(\cdot)$  is the sigmoid activation function,  $\mathcal{N}_k$  are the  $L$  nearest neighbours of node  $k$  based of positional Euclidean distance, and

$$\begin{aligned}
\gamma_{kl} &= \prod_{j \in \mathcal{N}_k \setminus l} (1 - \alpha_{kj}^{(t)}) \left\{ \sum_{m \in \mathcal{N}_k \setminus l} \frac{\alpha_{km}^{(t)}}{(1 - \alpha_{km}^{(t)})} \left[ (\beta_2 - \beta_1) \right. \right. \\
&- \left. \beta_2 \sum_{n \in \mathcal{N}_k \setminus l, m} \frac{\alpha_{kn}^{(t)}}{(1 - \alpha_{kn}^{(t)})} \right] + (\beta_1 - \beta_0) \left. \right\} + \mathbf{a}^T \mathbf{x}_k \\
&+ (4\alpha_{lk}^{(t)} - 2)\lambda + 2\alpha_{lk}^{(t)} (\boldsymbol{\eta}^T |\mathbf{x}_k - \mathbf{x}_l| + \boldsymbol{\nu}^T (\mathbf{x}_k \circ \mathbf{x}_l)). \tag{4.28}
\end{aligned}$$

## Chapter 5

# Summary and Discussion

### 5.1 Summary

Four novel methods to extract trees from volumetric CT data were presented in this thesis with a focus on segmenting airway trees.

In Chapter 2, modifications to an existing interactive method based on multiple hypothesis tracking (MHT) were presented. It was proposed to rank local hypotheses to quantify their relative significance; this enabled the method to overcome scale dependence. The bifurcation handling strategy was updated so that new branches inherited their parent's hypothesis tree instead of restarting from scratch, which allows the new branches to access more global information. The proposed changes turned the modified MHT method into an automatic one, capable of segmenting tree-structures. Experiments performed to extract airways from CT data and on coronary artery extraction CTA data, qualitatively showed these improvements to be useful. In both sets of experiments, the modified MHT method was used in automatic settings requiring just a single seedpoint per segmented structure. In the airway extraction experiments, the modified MHT method was shown to significantly improve upon the original MHT method used in non-interactive settings, region growing on intensity and probability images. With the coronary extraction method, again, it was shown that the modified MHT method showed significant improvements compared to the original MHT method and it ranked in the top 3 semi-automatic methods in the Coronary Artery challenge.

The MHT method in Chapter 2 was first of the methods investigated in this project with an objective of incorporating more global information when making local decisions. Remaining methods presented in Chapters 3 and 4 develop and build on this idea within the framework of probabilistic graphical models.

In Chapter 3, a probabilistic state-space model was introduced and Bayesian

smoothing was performed to extract airway branches. A two-step preprocessing procedure was used, to convert intensity images into probability maps using a trained voxel classifier which was followed by a multi-scale blob detection step, to obtain a sparse representation of possible airway regions in volumetric images. Linear Gaussian process and measurement models were proposed to perform Bayesian smoothing on these sparse representations yielding candidate branches. Each predicted branch comprised a set of nodes associated with multivariate Gaussian density estimates. The state space at each of these node comprised local scale, location and orientation in 3D. Airway branches were validated from the candidate branches by tuning a threshold on the total variance estimated from the individual Gaussian density estimates of all nodes in a branch. The resulting output from this method was a collection of disconnected branches. To demonstrate the usefulness of this method we compared its performance by merging its predictions with that of the comparing method; this revealed additional branches and terminal airways that were missed in the comparing method were detected by the proposed method.

As the output from the method in Chapter 3 was a set of candidate branches (not a connected tree), with sets of nodes, in Chapter 4 we formulated extraction of trees from these branches as a graph refinement task. Input graphs were constructed based on the output from the Bayesian smoothing method with a feature vector per node comprising mean and variance from the corresponding multivariate Gaussian densities. These nodes were connected to a predetermined number of nearest neighbours to obtain over-complete graphs to be used as input to the graph refinement methods.

The first graph refinement method proposed in Chapter 4 took up an approximate Bayesian inference approach. The posterior density over the subgraphs was approximated using a simpler, factorisable, density using mean-field approximation (MFA). Parameterised node and pairwise potentials were modeled to capture the behaviour of nodes of the subgraphs of interest. Based on this model, the update equations for the MFA iterations were worked out. By unrolling MFA iterations as feed-forward operations of a neural network, parameters of the MFA model were learnt using gradient descent; this connection was pointed out to be mean-field networks (MFNs) [Li and Zemel, 2014].

The MFN model required very little supervision as it was based on hand-crafted node and pairwise potentials. In order to investigate if the update equations implemented in the MFN model could be learnt from data in a supervised setting, a study using graph neural networks which are seen as generalisation of message passing algorithms was carried out.

In Chapter 4, the second method for graph refinement based on graph neural networks (GNNs) in a supervised learning setting was presented. By

jointly training an encoder-decoder pair using the training data, useful edge embeddings were learnt from which the probability of edge connections were predicted. Subgraphs corresponding to the connectivity in the centerline of interest were obtained by thresholding these edge probabilities. Performance of the MFN and GNN models were compared to each other and a baseline method. These experiments showed that both the proposed methods achieved significant improvement in their ability to detect more airway branches.

## 5.2 Discussion and conclusion

Large number of airway segmentation methods in literature start from initial seed points and take up a sequential approach by making local segmentation decisions [Lo et al., 2012, Pu et al., 2012]. Methods, such as region growing, that rely on making such local decisions are susceptible to occlusions and imaging artifacts, possibly, resulting in large portions of missing branches. Handling missing branches is all the more important when these methods are used to detect airways in relation to COPD diagnosis, as the small and terminal airways are the most affected [GOLD, 2017]. Exploratory methods that are not limited to segmenting branches from fixed seed points can alleviate this, for instance, by detecting branches from both directions. Another way of overcoming these occlusions can be by using more global (long-range) information. This has been attempted in [Bauer et al., 2015] using a graph based optimisation of likely local airway branches, and in [Meng et al., 2017] where U-Net based segmentation predictions were performed on volume patches that include a larger neighbourhood and tracking on top of it to obtain the final branches. Another class of commonly encountered methods is template-matching based such as [Born et al., 2009, Bauer et al., 2009], which use domain knowledge about airway morphology to some extent in constructing the tubular templates. As airways are regular tree structures and their evolution in the lungs is understood to a large extent [Weibel and Gomez, 1962, Kitaoka et al., 1999, Weibel, 2009], incorporating this information beyond the tubular templates can be beneficial. For instance, knowledge about the airway tree structure can also be used in the tree reconstruction procedure after obtaining interesting airway candidates. Consider the method in [Bian et al., 2018] that uses a random forest based voxel classifier to obtain an airway probability map. The final segmentation then is obtained by a simple thresholding operation; this step can be strengthened by using domain knowledge about airway trees. Lastly, uncertainty estimation of predictions can yield better insight into the model behaviour which is largely unexplored for airway segmentation tasks.

This thesis focused on addressing the aforementioned concerns by proposing

four airway segmentation methods in Chapters 2, 3 and 4 aimed to fulfil the following properties:

- Developing exploratory methods:

A key feature of all the four presented methods is their exploratory nature. The modified MHT method presented in Chapter 2, at the outset, is a sequential method that starts tracking from a seed point. However, it explores different global hypotheses in the hypothesis tree which enables it to overcome local occlusions. The probabilistic graphical model based methods in Chapters 3 and 4 do not start tracking from a single point like several other tree tracking algorithms [Lesage et al., 2009, Lo et al., 2012]. These graph-based methods can detect branches from both directions, and in cases with missing data due to occlusions segments of branches can still be detected. With the graph refinement approaches these partial segments can also be connected as presented in Chapter 4 .

- Utilising more global information in local decisions:

All the presented methods use information beyond the local neighbourhoods to make the segmentation decisions. Although differently when compared to other existing methods that use a larger neighbourhood like [Lo et al., 2009] where likely paths were probed within a feasible region from the current step and more recently with U-Net based methods where a larger volume patch is used for predicting airways [Meng et al., 2017, Juarez et al., 2018]. In addition to using a larger neighbourhood, the presented methods also use structured long range information to improve local decisions. The decisions in the MHT method in Chapter 2, are based on traversing the hypothesis tree. Depending on the depth of the hypothesis tree a local decision is based on a larger neighbourhood, also taking into account the history of decisions leading up to the current step. The Bayesian smoothing method in Chapter 3 extracts a branch by performing two pass Bayesian filtering; one forward pass and one backward pass [Särkkä, 2013]. This improves the range of information that is accessible to the method. Also, a gating procedure is utilised at each update step, as described in Section 3.2, to evaluate several candidate measurements instead of updating the prediction with the nearest measurement. The graph refinement methods proposed with MFN and GNN also base their local decisions based on a larger neighbourhood. The MFN model uses iterative message passing from neighbourhoods of nodes to predict local connection probabilities. In the GNN model the receptive field for nodes can be increased by stacking more GNN layers in the encoder allowing nodes to access information from a larger neighbourhood; in principle, it

is possible for a node in the GNN model to access information from the entire graph.

- Use of prior knowledge about the structures into models:

Tree structures like airways are consistent, to a large extent, in their anatomy across subjects [Kitaoka et al., 1999, Weibel, 2009]. Incorporating existing domain knowledge about such structures can be useful when performing segmentation. For instance, in a previous work in [Lo et al., 2010], the information that airways are paired with pulmonary vessels was used to improve airway segmentations based on their similarity in orientation. All the four methods presented in this thesis attempt to incorporate domain knowledge into the models to varying degrees. The MHT method in Chapter 2 uses templates reflecting the fact that airways are tubular structures as seen in Equation 2.1. Some of the MHT parameters such minimum and maximum radii were fixed, and ranges of few others constrained, based on airway morphology as seen in Table 2.1. In the Bayesian smoothing model in Chapter 3, the evolution of branches is captured in the motion model in Equation (3.3) to reflect the understanding that individual airway branches, between bifurcations, evolve in a smooth and almost linear fashion. With the MFN model presented in Chapter 4, the node and pairwise potentials in Equations (4.2) and (4.3) encode thorough domain knowledge relevant to the behaviour of nodes belonging to branches in an airway tree and also to distinguish false positive nodes. For instance, the choice of parameterising node degrees explicitly for  $D = [0, 1, 2]$  in Equation (4.2) was based on the understanding that the most commonly encountered nodes in such tasks would have either no edges ( $D = 0$ ), have one edge (root or terminal nodes) or two edges (one each to parent and a child). The choice to normalise the difference in spatial location of nodes with their average scale in Equation (4.3), as described in Section 4.3.1, was based on the fact that in our sprase representation the distance between nearest neighbours is larger for larger nodes than for smaller nodes. Even in the supervised learning setting of using GNNs in Chapter 4, domain knowledge is expressed as the particular choice of node features that are associated to nodes in the input graph.

- Estimation of prediction uncertainty:

Estimation of prediction uncertainty can help understand the models better and in some cases also improve the segmentations. Prediction uncertainty is innate to the probabilistic graphical model based methods in Chapters 3 and 4. In the Bayesian smoothing method in Chapter 3, the estimated

uncertainty was used to eliminate false positive branches from the predicted branches based on a threshold according to Equation (3.19). In the graph refinement setting in Chapter 4, the probability of edge connections is output from both MFN and GNN models. One can gain more insights into these models and their predictive capabilities either by visualising the predictions at different thresholds or performing a qualitative analysis based on these predicted probabilities [Nair et al., 2018]. Although not within the Bayesian setting, there is a very useful notion of uncertainty used in the MHT method in Chapter 2. One important modification proposed to the original MHT method was to use relative ranking of local hypotheses and to compute ranking based local hypothesis scores, according to Equation (2.6). Then, the global hypothesis scores were computed by averaging scores of the component local hypotheses, giving the global hypothesis scores a probabilistic interpretation. As the segmentation decision at each step was tied to a global hypothesis, uncertainty in these decisions can be quantified simply based on the global hypothesis score at that decision point.

In addition to yielding two effective tree reconstruction methods, the graph refinement methods presented in Chapter 4, provided interesting insights into the workings of GNNs and message passing algorithms. Classical approximate inference methods like MFA are iterative in nature. They fall into the class of message passing algorithms that have been widely used for inference in graphical models [Wainwright et al., 2008]. By unrolling MFA iterations as feed-forward operations of a neural network, it was pointed out in Chapter 4 that the parameters can be learnt using gradient descent, yielding the mean-field network (MFN) [Li and Zemel, 2014]. MFNs can be interpreted as structured neural networks, whose structure is based on an underlying graphical model. The graphical model itself encodes strong domain knowledge which is task dependent; this can also be seen as model specification bias. On the other hand, GNNs can learn task-specific messages from data in a supervised setting. GNNs can be seen as generalisations of message passing algorithms and have been shown to approximate the same marginal probabilities as message passing algorithms [Yoon et al., 2018]. But, these task-specific messages learnt by the GNN models are sensitive to the training data and thus introduce variance into the model. Beyond seeing GNNs as generalisations of MFNs and MFNs as constrained GNNs, it is also interesting to see the classic bias-variance trade off between these models. Relatively speaking, the MFN model has more bias and lower variance, whereas the GNN model has lower bias and higher variance.

Finally, the methods in this thesis were presented with a focus on segmenting airway trees. It must, however, be noted these are easily generalisable to



segmentation of other tree-structures encountered in both medical and non-medical volumetric images.

In conclusion, the four models presented in this thesis attempted to bridge a gap in the literature for tree-segmentation methods; especially airway segmentation methods. The proposed models were exploratory, used more global information, incorporated domain knowledge and estimated prediction uncertainty. Existing methods like [Bauer et al., 2015] are exploratory that uses a tree reconstruction approach on candidate branches obtained using tube detection filters. Methods like [Lo et al., 2009, Meng et al., 2017, Bian et al., 2018] have used larger neighbourhoods to make the segmentation decisions and shown it to be useful. Domain knowledge has been included in existing methods like in [Lo et al., 2010] where airway and vessel orientation similarity was used to improve airway segmentation, and in [Zhao et al., 2017] which uses tree statistics based prior for extracting airways. While these existing methods address some of the concerns addressed in this thesis separately, the models proposed in this thesis strived to address all these concerns simultaneously in varying degrees.

The most promising methods studied in this work would be the MFN and GNN models for graph refinement as they possess all the traits sought with this work to a large extent. However, due to the preprocessing pipeline followed in this work, they might not be immediately applicable in clinics or for broader studies. With some strategies recommended in Section 5.3 this can be alleviated. The recent advancements in using 3D CNNs for medical image segmentation [Litjens et al., 2017], have made models like U-Net ubiquitous [Çiçek et al., 2016, Juarez et al., 2018]. They are currently known to be memory intensive when dealing with volumetric images but there is no denying that CNNs are powerful feature extractors. On the other hand, in this work it has been shown that, given sufficiently expressive node features, the graph refinement models using MFN and GNN are reliable in extracting tree-structures by capturing long-range information while not being as resource hungry as 3D CNNs. The ideal confluence in the near future would be to use 3D CNNs locally in volumetric images to extract expressive features, and then build a graph refinement model either using MFNs or GNNs to get the best out of both these classes of models. The choice of using MFNs or GNNs for the graph refinement itself would depend on the extent of easily available domain knowledge that can be modelled versus the availability of sufficient labelled training data, respectively.

### 5.3 Future prospects

Several interesting and promising directions of research with the project could not be pursued due to several constraints; time being the primary one. A brief

overview of some of the shortcomings of this study and interesting directions for future investigation are presented here.

The primary scope of the methods presented in this thesis was segmentation of tree-structures from volumetric data. It had a focus on detection of airways. The next useful step is to perform measurements of airway lumen diameter and airway wall thickness using methods like [Petersen et al., 2014], based on the detections obtained from the proposed methods. Further, deriving biomarkers and correlating them with clinical outcomes can be most useful.

All the methods presented in this thesis employed a series of preprocessing steps starting with the conversion of intensity images into probability maps, based on the voxel classifier in [Lo et al., 2010]. With the advancement of new end-to-end learning based methods, a voxel classifier based on deep learning to obtain more useful preprocessing, as was recently attempted in [Juarez et al., 2018], can be interesting.

An end-to-end trainable model that uses dense volumetric data to construct sparse graphs dynamically with learnt features and uses one of the graph refinement methods in Chapter 4 to extract structures of interest can be most exciting. Local feature extraction could benefit from the capabilities of CNNs [Goodfellow et al., 2016], while global connectivity models can be incorporated using MFNs or GNNs. Early work in this direction for 2D vessel segmentation has been proposed in [Shin et al., 2018]. Such tandem models of 3D CNNs used along with MFNs can also open up further possibilities for interesting and more expressive models. One simple step could be to use CNNs in place of hand-crafted potential functions of Equations (4.2) and (4.3) and adapting the MFN model correspondingly. Further, as has been shown in [Lo et al., 2010], information from pulmonary vessels can be used to improve airway segmentation. Potential functions that employ CNNs to derive such joint airway-vessel modelling information within the MFN framework can be interesting.

# List of Publications

1. Raghavendra Selvan, Max Welling, Jesper H. Pedersen, Jens Petersen, and Marleen de Bruijne. “Mean field network based graph refinement with application to airway tree extraction.” 21st Conference on Medical Image Computing & Computer Assisted Intervention (MICCAI 2018), pp. 750-758, Cham. Springer International Publishing. <https://arxiv.org/abs/1804.03348>
2. Raghavendra Selvan, Thomas Kipf, Max Welling, Jesper H. Pedersen, Jens Petersen, and Marleen de Bruijne. “Extraction of Airways using Graph Neural Networks.” 1st Conference on Medical Imaging with Deep Learning (MIDL 2018), Amsterdam. <https://openreview.net/forum?id=rkn2fjjjG>
3. Raghavendra Selvan, Jens Petersen, Jesper H. Pedersen, and Marleen de Bruijne. “Extraction of airways with probabilistic state-space models and Bayesian smoothing.” In Graphs in Biomedical Image Analysis, Computational Anatomy and Imaging Genetics, MICCAI, 2017, pp. 53-63. Springer, Cham, <https://arxiv.org/abs/1708.02096>
4. Raghavendra Selvan, Jens Petersen, Jesper H. Pedersen, and Marleen de Bruijne. “Extraction of airway trees using multiple hypothesis tracking and template matching”. In The Sixth International Workshop on Pulmonary Image Analysis. MICCAI, 2016. <https://arxiv.org/abs/1611.08131>

## In preparation

5. Raghavendra Selvan, Jens Petersen, Jesper H. Pedersen, and Marleen de Bruijne. “Extracting Tree-structures in CT data by Tracking Multiple Statistically Ranked Hypotheses”. (2018) <https://arxiv.org/abs/1806.08981>

6. Raghavendra Selvan, Thomas Kipf, Max Welling, Jesper H. Pedersen, Jens Petersen, and Marleen de Bruijne. “Extracting Tree Structures using Mean-Field Networks and Graph Neural Networks”. (2018)

# Bibliography

- [Bar-Shalom et al., 2011] Bar-Shalom, Y., Willett, P. K., and Tian, X. (2011). *Tracking and data fusion*. YBS publishing Storrs, CT, USA:.
- [Bauer and Bischof, 2008] Bauer, C. and Bischof, H. (2008). Edge based tube detection for coronary artery centerline extraction. *The Insight Journal*.
- [Bauer et al., 2015] Bauer, C., Eberlein, M., and Beichel, R. R. (2015). Graph-based airway tree reconstruction from chest ct scans: Evaluation of different features on five cohorts. *IEEE Trans. Med. Imaging*, 34(5):1063–1076.
- [Bauer et al., 2009] Bauer, C., Pock, T., Bischof, H., and Beichel, R. (2009). Airway tree reconstruction based on tube detection. In *Proc. of Second International Workshop on Pulmonary Image Analysis*, pages 203–213.
- [Beal et al., 2003] Beal, M. J. et al. (2003). *Variational algorithms for approximate Bayesian inference*. university of London London.
- [Benmansour and Cohen, 2011] Benmansour, F. and Cohen, L. D. (2011). Tubular structure segmentation based on minimal path method and anisotropic enhancement. *International Journal of Computer Vision*, 92(2):192–210.
- [Bian et al., 2018] Bian, Z., Charbonnier, J., Liu, J., Zhao, D., Lynch, D., and van Ginneken, B. (2018). Small airway segmentation in thoracic computed tomography scans: a machine learning approach. *Physics in Medicine & Biology*, 63(15):155024.
- [Blackman, 2004] Blackman, S. S. (2004). Multiple hypothesis tracking for multiple target tracking. *IEEE Aerospace and Electronic Systems Magazine*, 19(1):5–18.
- [Born et al., 2009] Born, S., Iwamaru, D., Pfeifle, M., and Bartz, D. (2009). Three-step segmentation of the lower airways with advanced leakage-control. In *Proc. of Second International Workshop on Pulmonary Image Analysis*, pages 239–249.

- [Bronstein et al., 2017] Bronstein, M. M., Bruna, J., LeCun, Y., Szlam, A., and Vandergheynst, P. (2017). Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42.
- [Cetin et al., 2013] Cetin, S., Demir, A., Yezzi, A., Degertekin, M., and Unal, G. (2013). Vessel tractography using an intensity based tensor model with branch detection. *IEEE transactions on medical imaging*, 32(2):348–363.
- [Cetin and Unal, 2015] Cetin, S. and Unal, G. (2015). A higher-order tensor vessel tractography for segmentation of vascular structures. *IEEE transactions on medical imaging*, 34(10):2172–2185.
- [Charbonnier et al., 2017] Charbonnier, J.-P., Van Rikxoort, E. M., Setio, A. A., Schaefer-Prokop, C. M., van Ginneken, B., and Ciompi, F. (2017). Improving airway segmentation in computed tomography using leak detection with convolutional networks. *Medical image analysis*, 36:52–60.
- [Chung, 1997] Chung, F. R. (1997). *Spectral graph theory*. Number 92. American Mathematical Soc.
- [Çiçek et al., 2016] Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., and Ronneberger, O. (2016). 3D u-net: learning dense volumetric segmentation from sparse annotation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 424–432. Springer.
- [Cruz, 2007] Cruz, A. A. (2007). *Global surveillance, prevention and control of chronic respiratory diseases: a comprehensive approach*. World Health Organization.
- [Dijkstra, 1959] Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271.
- [Donohue and Ascoli, 2011] Donohue, D. E. and Ascoli, G. A. (2011). Automated reconstruction of neuronal morphology: an overview. *Brain research reviews*, 67(1-2):94–102.
- [Feuerstein et al., 2009] Feuerstein, M., Kitasaka, T., and Mori, K. (2009). Adaptive branch tracing and image sharpening for airway tree extraction in 3-d chest ct. In *Proc. Second International Workshop on Pulmonary Image Analysis*.
- [Florin et al., 2005] Florin, C., Paragios, N., and Williams, J. (2005). Particle filters, a quasi-monte carlo solution for segmentation of coronaries. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 246–253. Springer.

- [Frangi et al., 1998] Frangi, A. F., Niessen, W. J., Vincken, K. L., and Viergever, M. A. (1998). Multiscale vessel enhancement filtering. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 130–137. Springer.
- [Friman et al., 2010] Friman, O., Hindennach, M., Kühnel, C., and Peitgen, H.-O. (2010). Multiple hypothesis template tracking of small 3D vessel structures. *Medical image analysis*, 14(2):160–171.
- [Friman et al., 2008] Friman, O., Kühnel, C., and Peitgen, H.-O. (2008). Coronary centerline extraction using multiple hypothesis tracking and minimal paths. In *Proc MICCAI*, volume 42.
- [Gilmer et al., 2017] Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., and Dahl, G. E. (2017). Neural message passing for quantum chemistry. *arXiv preprint arXiv:1704.01212*.
- [GOLD, 2017] GOLD (2017). *Global Strategy for the Diagnosis, Management and Prevention of COPD, Global Initiative for Chronic Obstructive Lung Disease (GOLD) 2017*. GOLD.
- [Goodfellow et al., 2016] Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. (2016). *Deep learning*, volume 1. MIT press Cambridge.
- [Graham et al., 2010] Graham, M. W., Gibbs, J. D., Cornish, D. C., and Higgins, W. E. (2010). Robust 3-d airway tree segmentation for image-guided peripheral bronchoscopy. *IEEE transactions on medical imaging*, 29(4):982–997.
- [Hasegawa et al., 2006] Hasegawa, M., Nasuhara, Y., Onodera, Y., Makita, H., Nagai, K., Fuke, S., Ito, Y., Betsuyaku, T., and Nishimura, M. (2006). Airflow limitation and airway dimensions in chronic obstructive pulmonary disease. *American journal of respiratory and critical care medicine*, 173(12):1309–1315.
- [Hogg et al., 2013] Hogg, J. C., McDonough, J. E., and Suzuki, M. (2013). Small airway obstruction in copd: new insights based on micro-ct imaging and mri imaging. *Chest*, 143(5):1436–1443.
- [Hogg et al., 2017] Hogg, J. C., Paré, P. D., and Hackett, T.-L. (2017). The contribution of small airway obstruction to the pathogenesis of chronic obstructive pulmonary disease. *Physiological reviews*, 97(2):529–552.
- [Homann, 2007] Homann, H. (2007). Implementation of a 3D thinning algorithm. *Insight Journal*, 421.

- [Jaakkola and Jordan, 1998] Jaakkola, T. S. and Jordan, M. I. (1998). Improving the mean field approximation via the use of mixture distributions. In *Learning in graphical models*, pages 163–173. Springer.
- [Juarez et al., 2018] Juarez, A. G.-U., Tiddens, H., and de Bruijne, M. (2018). Automatic airway segmentation in chest ct using convolutional neural networks. In *Image Analysis for Moving Organ, Breast, and Thoracic Images*, pages 238–250. Springer.
- [Kipf et al., 2018] Kipf, T., Fetaya, E., Wang, K.-C., Welling, M., and Zemel, R. (2018). Neural relational inference for interacting systems. *International Conference on Machine Learning*.
- [Kipf and Welling, 2016] Kipf, T. N. and Welling, M. (2016). Variational graph auto-encoders. In *NIPS Bayesian Deep Learning Workshop*.
- [Kipf and Welling, 2017] Kipf, T. N. and Welling, M. (2017). Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*.
- [Kitaoka et al., 1999] Kitaoka, H., Takaki, R., and Suki, B. (1999). A three-dimensional model of the human airway tree. *Journal of Applied Physiology*, 87(6):2207–2217.
- [Kuo et al., 2017] Kuo, W., de Bruijne, M., Petersen, J., Nasserinejad, K., Ozturk, H., Chen, Y., Perez-Rovira, A., and Tiddens, H. A. (2017). Diagnosis of bronchiectasis and airway wall thickening in children with cystic fibrosis: Objective airway-artery quantification. *European Radiology*, pages 1–10.
- [Lee and Reeves, 2009] Lee, J. and Reeves, A. P. (2009). Segmentation of the airway tree from chest CT using local volume of interest. In *Proc. of Second International Workshop on Pulmonary Image Analysis*, pages 273–284.
- [Lee et al., 1994] Lee, T.-C., Kashyap, R. L., and Chu, C.-N. (1994). Building skeleton models via 3-d medial surface axis thinning algorithms. *CVGIP: Graphical Models and Image Processing*, 56(6):462–478.
- [Lesage et al., 2009] Lesage, D., Angelini, E. D., Bloch, I., and Funka-Lea, G. (2009). A review of 3D vessel lumen segmentation techniques: Models, features and extraction schemes. *Medical image analysis*, 13(6):819–845.
- [Lesage et al., 2016] Lesage, D., Angelini, E. D., Funka-Lea, G., and Bloch, I. (2016). Adaptive particle filtering for coronary artery segmentation from 3D ct angiograms. *Computer Vision and Image Understanding*, 151:29–46.



- [Ley-Zaporozhan et al., 2008] Ley-Zaporozhan, J., Ley, S., and Kauczor, H.-U. (2008). Morphological and functional imaging in copd with ct and mri: present and future. *European radiology*, 18(3):510–521.
- [Li et al., 2015] Li, Y., Tarlow, D., Brockschmidt, M., and Zemel, R. (2015). Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493*.
- [Li and Zemel, 2014] Li, Y. and Zemel, R. (2014). Mean-field networks. *ICML Workshop on Learning Tractable Probabilistic Models*.
- [Lindeberg, 1998] Lindeberg, T. (1998). Feature detection with automatic scale selection. *International journal of computer vision*, 30(2):79–116.
- [Lipworth et al., 2014] Lipworth, B., Manoharan, A., and Anderson, W. (2014). Unlocking the quiet zone: the small airway asthma phenotype. *The Lancet Respiratory Medicine*, 2(6):497–506.
- [Litjens et al., 2017] Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghahfarokian, M., van der Laak, J. A., Van Ginneken, B., and Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical image analysis*, 42:60–88.
- [Lo et al., 2010] Lo, P., Sporring, J., Ashraf, H., Pedersen, J. J., and de Bruijne, M. (2010). Vessel-guided airway tree segmentation: A voxel classification approach. *Medical image analysis*, 14(4):527–538.
- [Lo et al., 2009] Lo, P., Sporring, J., Pedersen, J. J. H., and de Bruijne, M. (2009). Airway tree extraction with locally optimal paths. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 51–58. Springer.
- [Lo et al., 2012] Lo, P., Van Ginneken, B., Reinhardt, J. M., Yavarna, T., De Jong, P. A., Irving, B., Fetita, C., Ortner, M., Pinho, R., Sijbers, J., et al. (2012). Extraction of airways from ct (EXACT’09). *IEEE Transactions on Medical Imaging*, 31(11):2093–2107.
- [McDonough et al., 2011] McDonough, J. E., Yuan, R., Suzuki, M., Seyednejad, N., Elliott, W. M., Sanchez, P. G., Wright, A. C., Geftter, W. B., Litzky, L., Coxson, H. O., et al. (2011). Small-airway obstruction and emphysema in chronic obstructive pulmonary disease. *New England Journal of Medicine*, 365(17):1567–1575.
- [Meng et al., 2017] Meng, Q., Roth, H. R., Kitasaka, T., Oda, M., Ueno, J., and Mori, K. (2017). Tracking and segmentation of the airways in chest ct using

- a fully convolutional network. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 198–207. Springer.
- [Milletari et al., 2016] Milletari, F., Navab, N., and Ahmadi, S.-A. (2016). V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *3D Vision (3DV), 2016 Fourth International Conference on*, pages 565–571. IEEE.
- [Nair et al., 2018] Nair, T., Precup, D., Arnold, D. L., and Arbel, T. (2018). Exploring uncertainty measures in deep networks for multiple sclerosis lesion detection and segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 655–663. Springer.
- [Nakano et al., 2000] Nakano, Y., Muro, S., Sakai, H., Hirai, T., Chin, K., Tsukino, M., Nishimura, K., Itoh, H., Paré, P. D., Hogg, J. C., et al. (2000). Computed tomographic measurements of airway dimensions and emphysema in smokers: correlation with lung function. *American journal of respiratory and critical care medicine*, 162(3):1102–1108.
- [Orlando and Blaschko, 2014] Orlando, J. I. and Blaschko, M. (2014). Learning fully-connected crfs for blood vessel segmentation in retinal images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 634–641. Springer.
- [Pal and Pal, 1993] Pal, N. R. and Pal, S. K. (1993). A review on image segmentation techniques. *Pattern recognition*, 26(9):1277–1294.
- [Pedersen et al., 2009] Pedersen, J. H., Ashraf, H., Dirksen, A., Bach, K., Hansen, H., Toennesen, P., Thorsen, H., Brodersen, J., Skov, B. G., Døssing, M., et al. (2009). The danish randomized lung cancer ct screening trial—overall design and results of the prevalence round. *Journal of Thoracic Oncology*, 4(5):608–614.
- [Petersen et al., 2014] Petersen, J., Nielsen, M., Lo, P., Nordenmark, L. H., Pedersen, J. H., Wille, M. M. W., Dirksen, A., and de Bruijne, M. (2014). Optimal surface segmentation using flow lines to quantify airway abnormalities in chronic obstructive pulmonary disease. *Medical image analysis*, 18(3):531–541.
- [Pham et al., 2000] Pham, D. L., Xu, C., and Prince, J. L. (2000). Current methods in medical image segmentation. *Annual review of biomedical engineering*, 2(1):315–337.

- [Pu et al., 2012] Pu, J., Gu, S., Liu, S., Zhu, S., Wilson, D., Siegfried, J. M., and Gur, D. (2012). Ct based computerized identification and analysis of human airways: A review. *Medical physics*, 39(5):2603–2616.
- [Reddi et al., 2018] Reddi, S. J., Kale, S., and Kumar, S. (2018). On the convergence of adam and beyond. *International Conference on Learning Representations*.
- [Reid, 1979] Reid, D. (1979). An algorithm for tracking multiple targets. *IEEE transactions on Automatic Control*, 24(6):843–854.
- [Roberts et al., 2000] Roberts, M., Reiss, M. J., and Monger, G. (2000). *Advanced biology*. Nelson Thornes.
- [Rosamond et al., 2008] Rosamond, W., Flegal, K., Furie, K., Go, A., Greenlund, K., Haase, N., Hailpern, S. M., Ho, M., Howard, V., Kissela, B., et al. (2008). Heart disease and stroke statistics—2008 update. *Circulation*, 117(4):e25–e146.
- [San Jose Estepar et al., 2008] San Jose Estepar, R., Reilly, J. J., Silverman, E. K., and Washko, G. R. (2008). Three-dimensional airway measurements and algorithms. *Proceedings of the American Thoracic Society*, 5(9):905–909.
- [Särkkä, 2013] Särkkä, S. (2013). *Bayesian filtering and smoothing*, volume 3. Cambridge University Press.
- [Scarselli et al., 2009] Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., and Monfardini, G. (2009). The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80.
- [Schaap et al., 2009] Schaap, M., Metz, C. T., van Walsum, T., van der Giessen, A. G., Weustink, A. C., Mollet, N. R., Bauer, C., Bogunović, H., Castro, C., Deng, X., et al. (2009). Standardized evaluation methodology and reference database for evaluating coronary artery centerline extraction algorithms. *Medical image analysis*, 13(5):701–714.
- [Selvan et al., 2018a] Selvan, R., Kipf, T., Welling, M., Pedersen, J. H., Petersen, J., and de Bruijne, M. (2018a). Extraction of airways using graph neural networks. *First International Conference on Medical Imaging with Deep Learning*.
- [Selvan et al., 2016] Selvan, R., Petersen, J., Pedersen, J. H., and de Bruijne, M. (2016). Extraction of airway trees using multiple hypothesis tracking and template matching. In *The Sixth International Workshop on Pulmonary Image Analysis*. CreateSpace Independent Publishing Platform.

- [Selvan et al., 2017] Selvan, R., Petersen, J., Pedersen, J. H., and de Bruijne, M. (2017). Extraction of airways with probabilistic state-space models and Bayesian smoothing. In *Graphs in Biomedical Image Analysis, Computational Anatomy and Imaging Genetics*, pages 53–63. Springer.
- [Selvan et al., 2018b] Selvan, R., Petersen, J., Pedersen, J. H., and de Bruijne, M. (2018b). Extracting tree-structures in ct data by tracking multiple statistically ranked hypotheses. *arXiv preprint arXiv:1806.08981*.
- [Selvan et al., 2018c] Selvan, R., Welling, M., Pedersen, J. H., Petersen, J., and de Bruijne, M. (2018c). Mean field network based graph refinement with application to airway tree extraction. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, pages 750–758, Cham. Springer International Publishing.
- [Setarehdan and Singh, 2012] Setarehdan, S. K. and Singh, S. (2012). *Advanced algorithmic approaches to medical image segmentation: state-of-the-art applications in cardiology, neurology, mammography and pathology*. Springer Science & Business Media.
- [Shin et al., 2018] Shin, S. Y., Lee, S., Yun, I. D., and Lee, K. M. (2018). Deep vessel segmentation by learning graphical connectivity. *arXiv preprint arXiv:1806.02279*.
- [Sørensen et al., 2011] Sørensen, L., Lo, P., Dirksen, A., Petersen, J., and De Bruijne, M. (2011). Dissimilarity-based classification of anatomical tree structures. In *Biennial International Conference on Information Processing in Medical Imaging*, pages 475–485. Springer.
- [Türetken et al., 2016] Türetken, E., Benmansour, F., Andres, B., Glowacki, P., Pfister, H., and Fua, P. (2016). Reconstructing curvilinear networks using path classifiers and integer programming. *IEEE transactions on pattern analysis and machine intelligence*, 38(12):2515–2530.
- [van Ginneken, 2017] van Ginneken, B. (2017). Fifty years of computer analysis in chest imaging: rule-based, machine learning, deep learning. *Radiological physics and technology*, 10(1):23–32.
- [Van Rikxoort and Van Ginneken, 2013] Van Rikxoort, E. M. and Van Ginneken, B. (2013). Automated segmentation of pulmonary structures in thoracic computed tomography scans: a review. *Physics in Medicine & Biology*, 58(17):R187.

- [Vos et al., 2016] Vos, T., Allen, C., Arora, M., Barber, R. M., Bhutta, Z. A., Brown, A., Carter, A., Casey, D. C., Charlson, F. J., Chen, A. Z., et al. (2016). Global, regional, and national incidence, prevalence, and years lived with disability for 310 diseases and injuries, 1990–2015: a systematic analysis for the global burden of disease study 2015. *The Lancet*, 388(10053):1545–1602.
- [Wainwright et al., 2008] Wainwright, M. J., Jordan, M. I., et al. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305.
- [Weibel, 2009] Weibel, E. R. (2009). What makes a good lung. *Swiss Med Wkly*, 139(27-28):375–386.
- [Weibel and Gomez, 1962] Weibel, E. R. and Gomez, D. M. (1962). Architecture of the human lung: Use of quantitative methods establishes fundamental relations between size and number of lung structures. *Science*, 137(3530):577–585.
- [Wielpütz et al., 2013] Wielpütz, M. O., Eichinger, M., Weinheimer, O., Ley, S., Mall, M. A., Wiebel, M., Bischoff, A., Kauczor, H.-U., Heussel, C. P., and Puderbach, M. (2013). Automatic airway analysis on multidetector computed tomography in cystic fibrosis: correlation with pulmonary function testing. *Journal of thoracic imaging*, 28(2):104–113.
- [Wiemker et al., 2009] Wiemker, R., Bülow, T., and Lorenz, C. (2009). A simple centricity-based region growing algorithm for the extraction of airways. In *Proc. Second International Workshop on Pulmonary Image Analysis (MICCAI)*, pages 309–314. Citeseer.
- [Wink et al., 2004] Wink, O., Niessen, W. J., and Viergever, M. A. (2004). Multiscale vessel tracking. *IEEE Transactions on Medical Imaging*, 23(1):130–133.
- [Yang et al., 2012] Yang, G., Kitslaar, P., Frenay, M., Broersen, A., Boogers, M. J., Bax, J. J., Reiber, J. H., and Dijkstra, J. (2012). Automatic centerline extraction of coronary arteries in coronary computed tomographic angiography. *The international journal of cardiovascular imaging*, 28(4):921–933.
- [Yedidya and Hartley, 2008] Yedidya, T. and Hartley, R. (2008). Tracking of blood vessels in retinal images using kalman filter. In *Computing: Techniques and Applications, 2008. DICTA’08. Digital Image*, pages 52–58. IEEE.
- [Yoon et al., 2018] Yoon, K., Liao, R., Xiong, Y., Zhang, L., Fetaya, E., Urtasun, R., Zemel, R., and Pitkow, X. (2018). Inference in Probabilistic Graphical

Models by Graph Neural Networks. *International Conference on Learning Representations Workshop Track*.

- [Zhang et al., 2001] Zhang, Y., Brady, M., and Smith, S. (2001). Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm. *IEEE transactions on medical imaging*, 20(1):45–57.
- [Zhao et al., 2017] Zhao, M., Miles, B., and Hamarneh, G. (2017). Leveraging tree statistics for extracting anatomical trees from 3D medical images. In *Computer and Robot Vision (CRV), 2017 14th Conference on*, pages 131–138. IEEE.
- [Zheng et al., 2013] Zheng, Y., Tek, H., and Funka-Lea, G. (2013). Robust and accurate coronary artery centerline extraction in cta by combining model-driven and data-driven approaches. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 74–81. Springer.