2008

# Improved association graph matching of intra-patient airway trees

Shalmali Vidyadhar Bodas
*University of Iowa*

# IMPROVED ASSOCIATION GRAPH MATCHING OF INTRA-PATIENT AIRWAY TREES

by

Shalmali Vidyadhar Bodas

A thesis submitted in partial fulfillment of the
requirements for the Master of Science
degree in Biomedical Engineering
in the Graduate College of
The University of Iowa

December 2008

Thesis Supervisor: Associate Professor Joseph M. Reinhardt

Graduate College
The University of Iowa
Iowa City, Iowa

CERTIFICATE OF APPROVAL

---

MASTER'S THESIS

---

This is to certify that the Master's thesis of

Shalmali Vidyadhar Bodas

has been approved by the Examining Committee for the thesis requirement for the Master of Science degree in Biomedical Engineering at the December 2008 graduation.

Thesis Committee: _____
Joseph M. Reinhardt, Thesis Supervisor

_____
Edwin L. Dove

_____
Eric A. Hoffman

To my beloved late grandmom Usha, who will always be a source of inspiration

## ACKNOWLEDGEMENTS

Pursuing my Master's had been my aim after I graduated from college. Coming from a middle-class family in India, it was definitely not an easy decision to make; especially with no financial aid available at the beginning of the program. Nonetheless, due to the strong, incessant support of my parents Vaijayanti and Vidyadhar Bodas and my maternal uncle Mohan Tilak, accompanied by the confidence that they showed in me has enabled me to make my dream come true today. I would like to take this opportunity to express my sincere gratitude to my family members without whose constant support and guidance I would not have been where I am today. I would also like to thank my husband Tanmay who has stood by me through thick and thin and given me all the support and encouragement I needed.

I am grateful to many people from my University for their help during these two years. I would like to thank my research advisor Prof. Joseph Reinhardt for his generosity in funding me during my studies as well as for his constant guidance, support and trust during my recent post-surgery period. I would also like to express my gratitude to Dr. Geoffrey McLennan, who shared his expertise with me and helped me in establishing the ground truth data. I am grateful to Dr. Eric Hoffman,the principal investigator of the NIH grant HL064368 that supported this work, for providing the data. I would like to thank the Department secretaries Lorena and April for their help. Last, but not the least, I would like to thank my friends and labmates Kai Ding, Sudharshan Bommu, Sangyeol Lee, Matt Ternus for their help and friendship.

This work was supported in part by grants HL079406 and HL064368 from the National Institutes of Health.

## ABSTRACT

Pulmonary diseases are frequently associated with changes in lung anatomy. These diseases may change the airway, vessel and lung tissue properties. In order to evaluate the lung in a longitudinal study, a stable reference system is required to identify corresponding parts of the lung. The structure of the airway tree can be used to repeatedly identify the regions of interest. In this study, an improved method for matching of intra-patient airway trees was proposed and evaluated. The association graph method proposed by Pelillo et al. [17, 18] matches free and rooted trees by detecting the maximal sub-tree isomorphism. Tschirren et al. implemented this approach for labeling and matching of human airway trees and reported 92.9% matching accuracy which is the highest among existing methods. However we recognized a few shortcomings of this method. When we tested it on seven normal human cases, we observed that successful matching relies heavily on the accurate labeling of main branchpoints in the trees. Incorrect labeling of main branch points or failure in labeling results in failure to match that branch point. Such matching errors may eventually propagate to sub-trees.

To improve the matching performance, we propose to make matching independent of labeling as well as improve association graph by adding constraint of path-length along with the existing constraints. Furthermore, we would like to redefine the incorrect matches as those matches which are mismatched as well as those that are missed by the matching algorithm. Our results for a total of 27 cases show a significant improvement in accuracy. The accuracy calculated as per the convention without accounting for the branchpoint pairs missed by the algorithm is 92.19%

whereas the accuracy calculated as per our definition is 73.98%, with runtime in the range of 0.01-262.56 sec (average runtime is 25.14 sec). We thus propose an improved association graph method which is efficient in matching intra-patient airway trees with good accuracy and within a reasonable time.

## TABLE OF CONTENTS

## LIST OF TABLES

## LIST OF FIGURES

# CHAPTER 1
# INTRODUCTION

## 1.1 Specific Aims

A lung disease is any disease or disorder where the lung is impaired. According to the Lung Disease Data 2008 collected by the American Lung Association, almost 400,000 Americans die from lung diseases every year and lung disease is the number three killer in the United States, responsible for one in six deaths. The survey also shows that more than 35 million Americans have chronic lung diseases [1]. As a result, the use of medical imaging for diagnosis, evaluation and treatment of lung diseases has been gaining importance recently. Lung diseases are frequently associated with changes in the lung anatomy. In order to evaluate the progression of or the effect of treatment on lung diseases, it is important to reproducibly identify the same region of the lung tissue across scans taken over a period of time. Thus, aligning of the images becomes crucial for objective evaluation and understanding. While dealing with organs such as the lungs, which expand and contract all the time, it also becomes necessary to consider the changes in anatomy occurring over a period of time due to the respiratory cycle, heart beats, etc. For this purpose, it is important to have some stable underlying reference system or landmarks. Among the popular techniques used for registration (alignment) of such images, the landmark-based approach involves identifying the landmarks in the structure and matching the corresponding landmarks in two images. The selection of landmarks is done so that they are stable and easily recognizable. The airway tree within the lung provides such a reference system that can be used to identify identical regions of lung tissue. Thus, matching of airway trees

can be used to track changes in the lung across longitudinal studies and to compare anatomy across individuals. In our research, we propose a method for intra-patient airway tree matching based on association graph technique. Our method differs from the previous one proposed by Tschirren et al [23, 24]. It works independent of the anatomical labels of the airway trees and uses an association graph to match the entire tree, thus overcoming some of the problems associated with the previous label-dependent approach.

# CHAPTER 2
# BACKGROUND AND SIGNIFICANCE

## 2.1 Pulmonary anatomy and physiology

The respiratory system is a group of organs and tissues that enable us to breathe. The lungs are the main organs of the respiratory system and the sites of gas exchange where air is inhaled to absorb oxygen and exhaled to remove carbon dioxide. During respiration, the air enters the body through the nose or mouth, travels down to the larynx and then to the trachea, which divides into two bronchi. The air travels through the left and the right main bronchi which further divide into smaller bronchi and bronchioles and terminates in tiny sacs called alveoli. These sacs are the main sites of gas exchange. Figure 2.1 shows the human respiratory system.

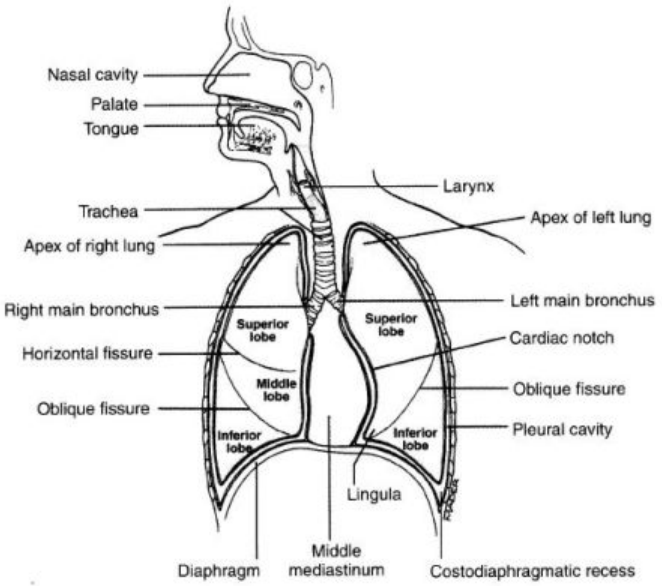



Figure 2.1: Human respiratory system. Figure from [13].

During respiration, the lungs expand and contract and hence the lung tissue undergoes continuous deformation. In patients with lung diseases such as asthma, emphysema, cystic fibrosis, etc. the lung mechanics are altered owing to changes in lung anatomy. Imaging techniques such as the chest CT scan are often used for the diagnosis of lung diseases and evaluation after treatment. These image-based evaluations may involve matching or registration of the scans taken at different times or at different lung volumes so that identical parts of the lung can be detected. Since the lung structure is prone to deformation and anatomical changes due to disease, it is necessary to have some reference to match the identical parts. This can be done via matching corresponding landmarks in the two scans.

The airway tree exhibits quite a similar pattern in branching across humans. These similarities can be observed up to the branch points where the lung divides into the sub-lobes. See Figure 2.2. This includes the 33 anatomically named segments that are commonly used in bronchoscopy. See Figure 2.3. Beyond these points, the pattern becomes more random and unpredictable. The airway tree within the lung can be used as a reference system for matching. Thus, by matching airway trees, we can identify corresponding branchpoints within the tree and hence identify the lobes and sub-lobes of the lung.

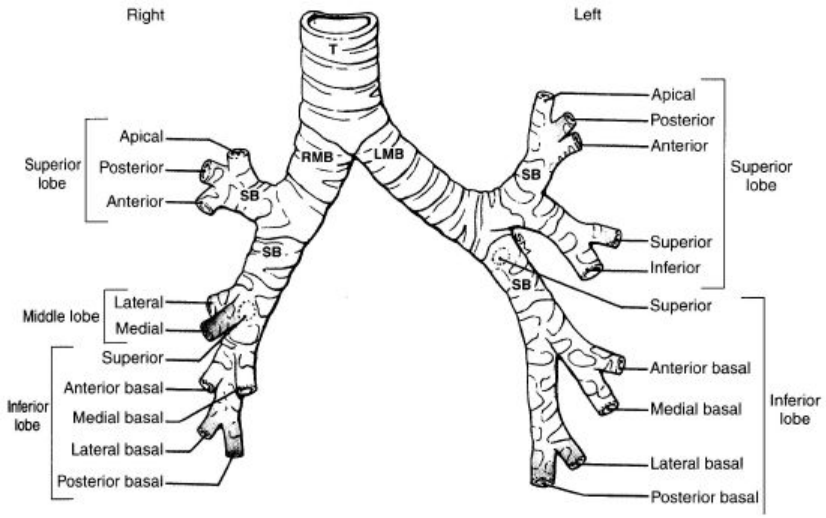



Figure 2.2: Typical branching pattern of human airway tree. Figure from [13].

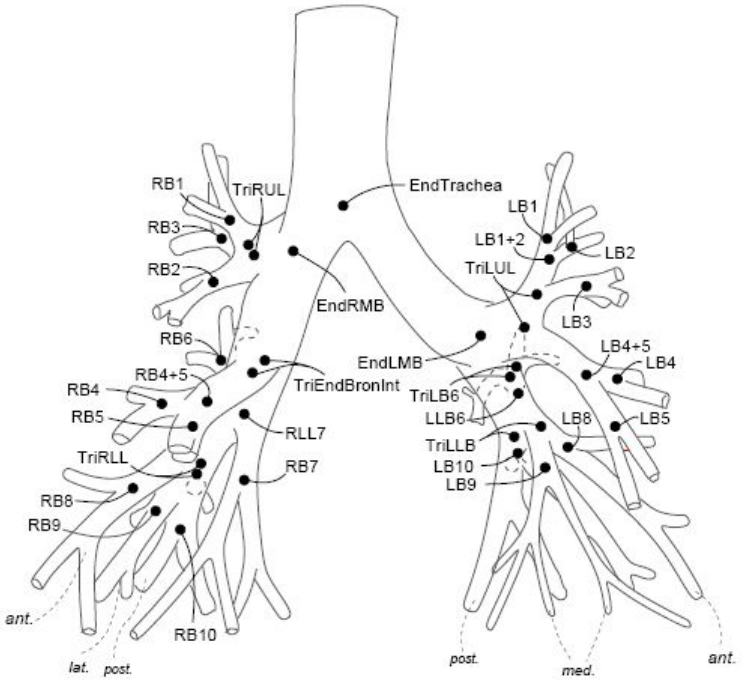



Figure 2.3: Nomenclature (Labels) of a typical human airway tree. Figure from [3].

## 2.2 Previous Work

Pisupati et al. [19, 20] proposed to solve the rooted binary tree isomorphism problem by designing a dynamic programming algorithm which incorporates topology and geometric heuristics. A rooted binary tree is a tree with a common reference node (root) having two branches per node. (For definition of tree isomorphism, please see Chap 3, Section 3.3.2 on page 16.) Since this approach uses similarity of geometrical features, its success depends largely on the similarity of the two trees to be matched. It is not tolerant to relatively big geometrical changes produced due to respiration, breathing etc. Also, it was tested on relatively simple airway tree of a dog with limited number of segments. Park [16] presented a tree-matching method based on an association graph. His method was applied to phantom data only and was not able to deal with false branches. Pelillo et al. [17, 18] proposed the association graph method for matching free and rooted trees by detecting the maximal sub-tree isomorphism. (For definition of tree isomorphism, see Chap 3, Section 3.3.2 on page 16.) The trees were treated as graphs with branchpoints as vertices and tree segments as the edges. The association graph was defined as a structure that consists of a vertex for every possible pair of vertices in the two graphs to be matched. Also, two vertices in the association graph were connected by an edge if and only if the consistency constraints were met. The node assignment were considered to be consistent if the topological relationship of the two nodes was equal. In the derived association graph, maximal clique was detected by applying the payoff-monotone dynamics from evolutionary game theory on a continuous formulation of the problem obtained by the Motzkin-Straus theorem. (Payoff monotone-dynamics is a class of simple dynamical systems recently developed and studied in evolutionary game theory [17].)

Tschirren et al. [23, 24] proposed an algorithm for matching and labeling of human airway trees. Their matching algorithm was based on an association graph as well. By exploiting the hierarchical information, matching was done on a sub-tree level by finding the major branchpoints using labels and then matching the sub-tree beneath them. This approach when tested on 17 human cases resulted in an average accuracy of 92.9% with a runtime of 1 to 3 seconds for trees with 200 to 300 branchpoints on a 1.2 GHz processor. However, this method failed to achieve one-to-one matching. Furthermore, accuracy was calculated without considering the matches that were present in ground truth but missed by the algorithm. Since this approach depends on the labels given to the airway trees to be matched, it fails when there are no branch labels in the trees and also tends to introduce errors in matching if the trees are incorrectly labeled. Another approach based on association graph proposed by Metzen et al. [12] used an enhanced association graph and tried to match the entire tree in one step. The algorithm was tested on bronchial trees and liver portal veins. It is shown that the algorithm achieved an accuracy of at least 84%. However, the major drawback of this approach was that the number of matches achieved highly exceeded the number of matches present in the ground truth. Also, good results were achieved only for typical examples of vessel trees. Graham et al. [9, 8] proposed a model-based approach where the extracted trees were assumed to arise from an initially unknown common tree structure corrupted by a sequence of modeled topological deformations. A novel mathematical framework to incorporate this model into the matching problem was implemented using a dynamic programming algorithm. The algorithm was tested on only one human dataset and the runtime for the matching of 141-341 vertices on a 3.4 Ghz processor was given as 5 seconds. Recently, Lohe

et al. [11] proposed a hierarchical tree search algorithm to compute the matching between branchpoints of anatomical trees. When tested on 11 datasets, it achieved an average accuracy of 80.9% with a runtime of 1-45 seconds for trees with 50-700 vertices. Among other non graph-based approaches, Kaftan et al. [10] proposed a method which matches complete path from root node to terminal node (leaf) and does not rely on branchpoint to branchpoint matching. It was tested on 10 human datasets and depending on the features chosen, matched an average of 87% paths correctly. Bulow et al. [5] introduced and compared two approaches to tree matching using features like 3D shape context and statistical moments of the local point distribution. The method was tested on seven scans, with one completely segmented tree being used as the model and the remaining six tree centerlines matched to it using the proposed shape features. This method correctly labeled an average of 69% and 40% trees with shape context and statistical moment feature. In another approach, Tang et al. [22] proposed an algorithm based upon minimization of tree edit distance between two vascular trees. The algorithm was tested on two datasets of cerebral vascular trees with more than 30 branches and matched only the major branchpoints correctly which were then verified by visual inspection. Charnoz et al. [7, 6] proposed a tree matching algorithm for intra-patient hepatic vascular system registration based on a set of matching hypotheses which was updated to keep the best matches. The vascular systems were segmented from CT images acquired at different time and then modeled as trees. This algorithm was validated on a large synthetic database, required a comparatively larger runtime of about 10 mins to register a tree with 380 nodes on a 1 GHz processor and achieved a sensitivity greater than 90%.

# CHAPTER 3
# MATERIALS AND METHODS

## 3.1 Overview

Amongst all the existing approaches to match airway trees, the association graph method proposed by Tschirren et al. [23, 24] has shown to be the most promising one with the maximum documented matching accuracy of 92.9%. This method consists of two steps - labeling the airway trees using a reference tree and matching the labeled airway trees. (See Figure 3.1.) The matching step can be further decomposed into two steps -

1. Matching the main branchpoints by identifying the common labels
2. Matching the sub-tree underneath the main branchpoints by using an association graph

Although the results are impressive, this method suffers from certain shortcomings, such as a dependency on the branch labels in the airway trees. This algorithm relies on accurate labeling of the two airway trees to be matched. Hence an error in labeling is likely to propagate and affect the sub-tree matching. Our observations show that mislabeled trifurcations lead to mismatches further into the sub-tree. In order to quantify how the labels affect matching, we tested this method with seven normal human datasets. We observed that the matching accuracy was much lower than the projected value. We were also able to identify some of the common mislabeled branchpoints, a majority of them being the trifurcations. (See Chap 4 Figure 4.1 on page 32).

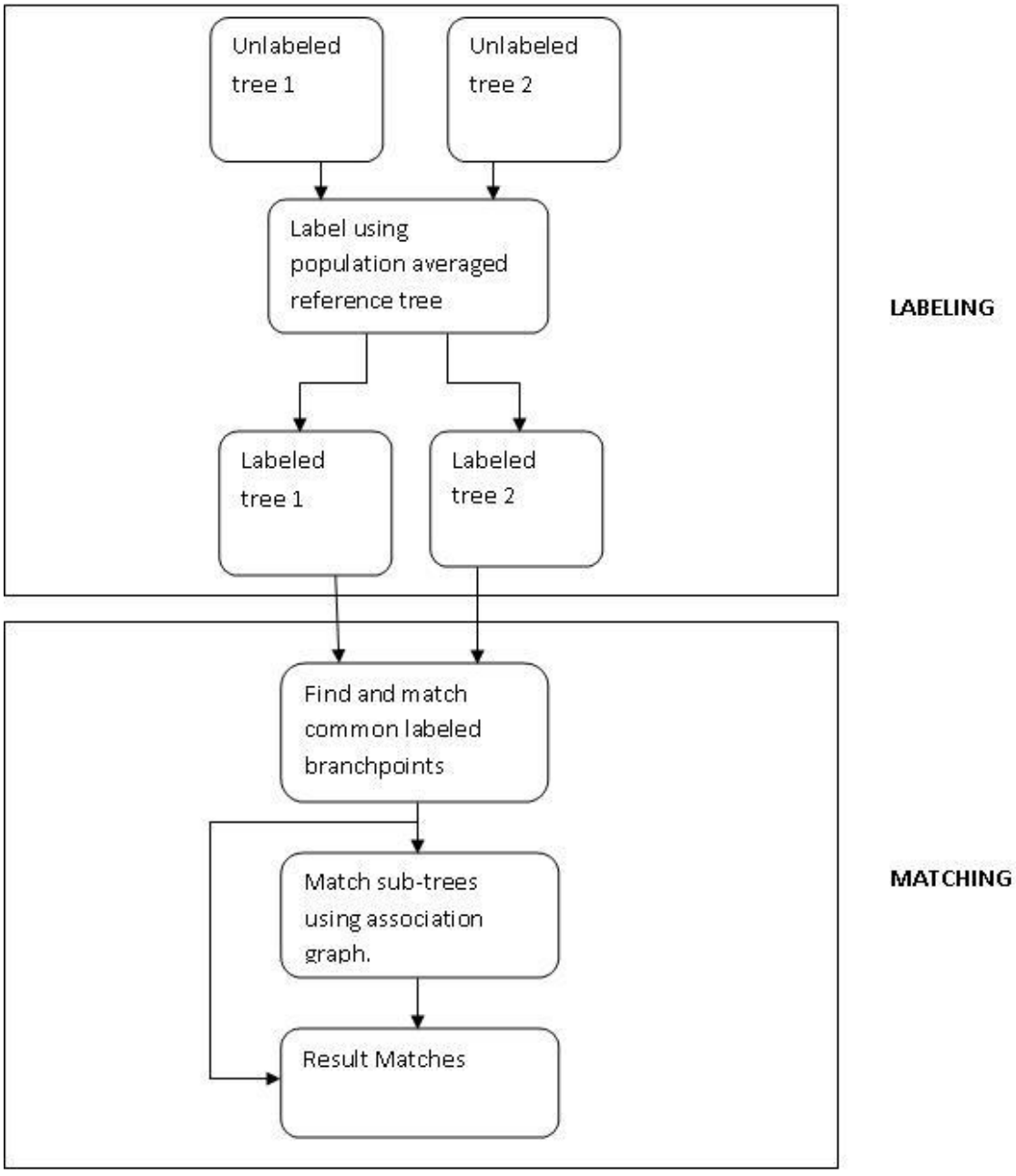



Figure 3.1: Previous Label-dependent approach as proposed by Tschirren et al [23, 24].

It was observed that if the algorithm missed labeling a branchpoint in either of the trees, there would be no match for that branchpoint and the sub-tree beneath it would not be discovered either. It was seen that a number of matches present in the hand-matched or the ground truth data were missing in the results produced by the algorithm. In our method, we aim to reduce these dependencies by not only improving the association graph but also by eliminating the initial label-wise matching step and assigning it via the association graph. (See Figure 3.3.) Moreover, in our approach, we choose to define the matching accuracy such that it considers those matches which are present in hand-matched data (ground truth) but are absent in the result produced by the algorithm.

*From here on, we refer the matches which are present in hand-matched data (ground truth) but are absent in the result produced by the algorithm as "missed matches".*

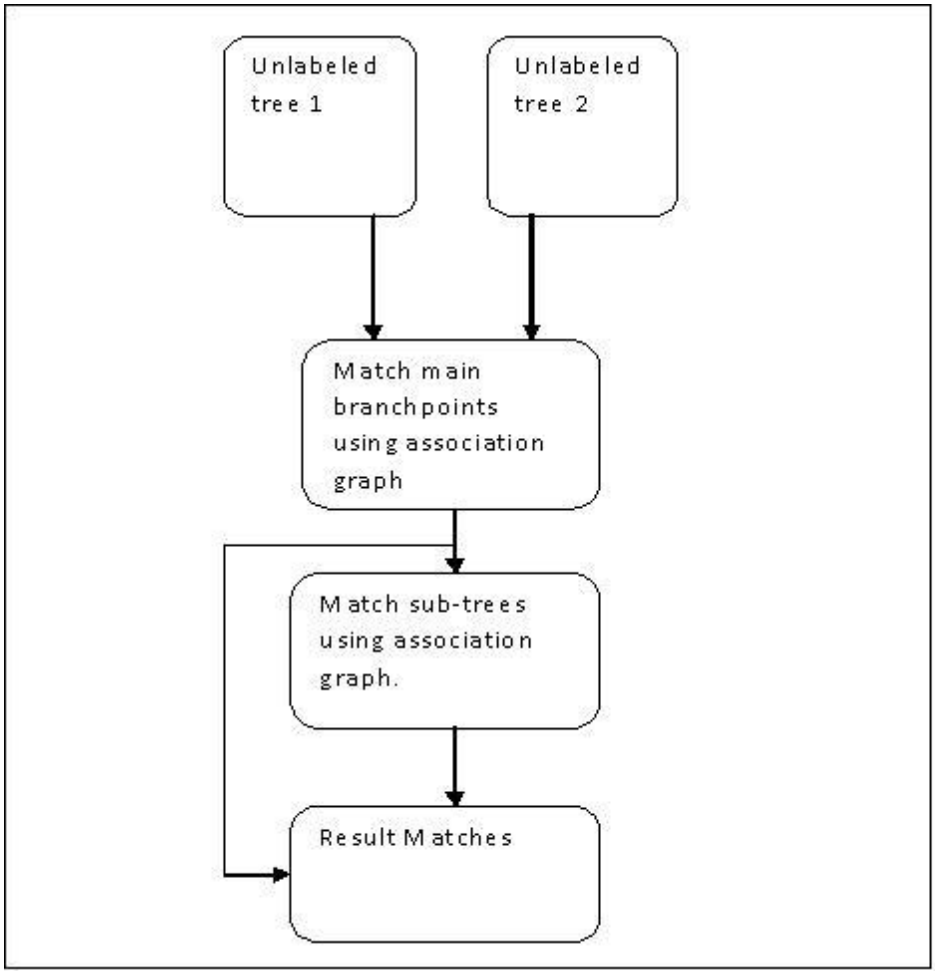



Figure 3.2: New Label-Independent approach.

## 3.2 Data Acquisition and Preprocessing

The subjects chosen for the study were classified into two groups: one group consisted of healthy individuals with no known lung disease and the second group consisted of those previously clinically diagnosed with interstitial lung disease as categorized by having one or more of these textural changes show up on a previous CT scans- ground glass, emphysema-like, bronchovascular, honeycombing, and nodular. The data was acquired using a Siemens Sensation 64 multi-detector CT (MDCT) scanner and the same protocol was followed for all the subjects. The scans were volume scans acquired at resting exhalation (functional residual capacity (FRC)) and full inspiration (total lung capacity (TLC)) with a pitch of 1, slice collimation 0.6 mm, rotation speed 0.5 sec, slice thickness 0.75 mm, increment 0.5 mm, 120 kV, 75 mAs and kernel B50f. The scans were acquired with a reconstruction matrix of $512 \times 512$ and reconstruction slice width of 0.5 mm. The acquired CT scans were then segmented and skeletonized using Pulmonary Workstation [2] to generate XML files. The input to the program were XML files generated from two input trees. All the data was gathered using a protocol approved by the University of Iowa Institutional Review Board.

## 3.3 Method

### 3.3.1 Overview

Our method is based on matching with an association graph. An association graph is an intermediate data structure built from two graphs to be matched. Airway trees can be depicted as directed acyclic graphs [24] with branchpoints as the vertices and the segments as the edges. See figure 3.4 for an example.

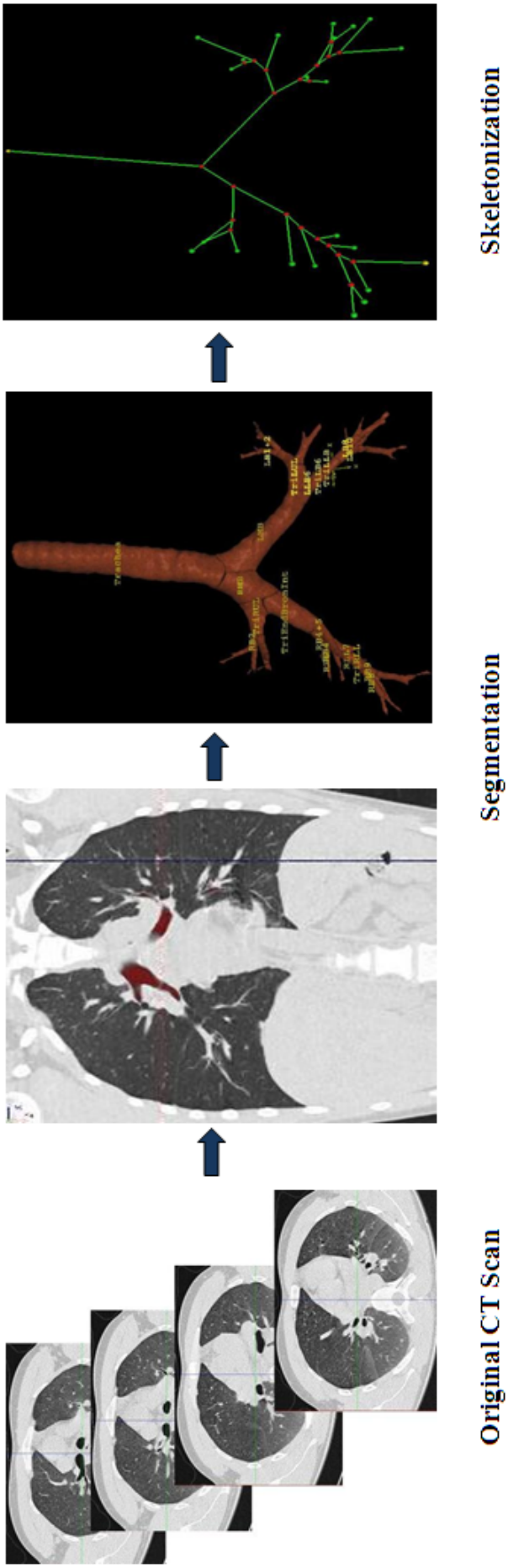



Figure 3.3: Data pre-processing steps

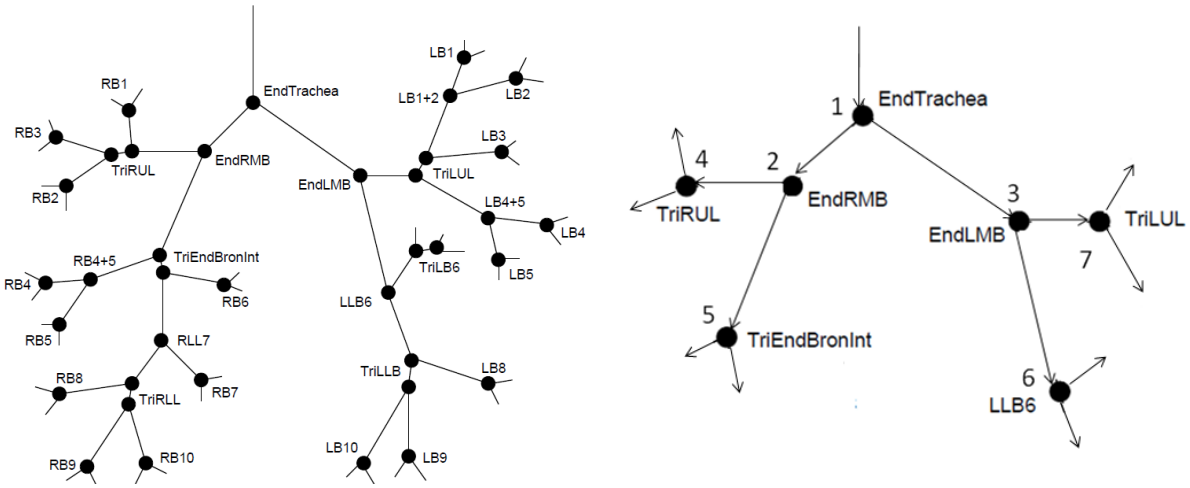



Figure 3.4: Fig a. depicts a typical skeletonized human tree.(Figure taken from [24].) Fig b. shows representation of a part of the tree as a directed acyclic graph.

The problem of airway tree matching can then be considered as the problem for finding the maximum tree-isomorphism (For definition, please see section 3.3.2) and hence the well known graph algorithm for finding maximum clique can be applied [23, 24]. However, in computational complexity theory, the clique problem is graph-theoretic NP-complete. This means that no fast solution to such problems is known; that is, the time required to solve the problem using any currently known algorithm increases very quickly as the size of the problem grows. In an effort to reduce the computing time, we would like to reduce the size of the datasets. As done in [23, 24], the time can be reduced if the main branchpoints are labeled, thus dividing the tree into smaller sub-trees. However, in order to reduce the dependency on the labels, we chose to do the assignments of main branchpoints as well as that of the sub-trees via association graph. This increases our computational time but gives more accurate results.

Thus, our algorithm consists of two steps-

1. Matching the main branchpoints using an association graph

2. Matching sub-trees using an association graph

### 3.3.2 Association graph and maximum clique-

Notations and definitions

Let $G = (V, E)$ be a graph where $V$ is a set of nodes (vertices) and $E$ is a set of edges. As per the definitions of graph theory, the order of $G$ is the number of nodes in $V$, while the size of $G$ is the number of edges in E. Two nodes $u, v$ are said to be adjacent if there is an edge connecting them. A rooted graph is the one which has a distinguished node called the root. In a rooted graph, the length of the path is the sum of the path lengths of all nodes lying within that path and measured from the root. The distance between two nodes $u, v$ is the length of the shortest path joining the two nodes. An association graph is an auxiliary data structure produces from two relational structure to be matched and a clique is a collection of vertices, each joined to the other by an edge [4].A vertex in the association graph represents a possible pair of vertices, with one vertex from each graph to be matched. Let $G_1$=($V_1$,$E_1$) and $G_2$=($V_2$,$E_2$) be the two graphs (alternatively trees) to be matched. Then *the bijection* $\phi : H1 \rightarrow H2$*, with* $H1 \subseteqq V1$ *and* $H2 \subseteqq V2$ *is the sub-tree isomorphism if it preserves both the adjacency and relationships between the nodes and the connectedness of the matched subgraphs. A sub-tree isomorphism is* maximal *if there is no other sub-tree isomorphism* $\acute{\phi} : \acute{H}1 \rightarrow \acute{H}2$ *with H1 a strict subset of* $\acute{H}1$ *, and* maximum *if H1 has largest cardinality. The maximal (maximum) sub-tree isomorphism problem is to find a maximal (maximum) sub-tree isomorphism between two trees* [17]. The association

graph $G_{ag} = (V_{ag}, E_{ag})$ will consist of $V_1$ x $V_2$ vertices. Two vertices in $G_{ag}$ are connected by an edge if and only if the corresponding vertices in $G_1$ and $G_2$ satisfy all the constraints imposed on them while matching. The maximum clique in the association graph is the maximum number of vertices in the association graph that are connected by an edge and corresponds to maximal sub-tree isomorphism. Since every vertex in the association graph corresponds to a pair of vertices from $G_1$ and $G_2$, every vertex in the maximum clique also corresponds to a pair of matching vertices in $G_1$ and $G_2$. The addition of edges in the association graph can be restricted by imposing several constraints. In his work, Pelillo [17] used path-length generation number as a constraint for adding edges in the association graph. However, this method works only if the two trees to be matched are topologically identical. Tschirren et al. [23, 24] used more number of constraints including topological distance, hierarchical information and geometric measures while adding edges to the association graph. In our method, we add one more geometric constraint of path length, in addition to the earlier ones.

### 3.3.3 Rigid registration

Prior to starting the matching process, we need to register the two input trees in order to bring them in the same coordinate system. Registration can be performed by using the characteristic first three main branchpoints viz. End Trachea (Carina), End LMB and End RMB (See Chap 2 Figure 2.3) in the two trees. Rigid registration of these points involves translation of End Trachea of the second tree to align with the first one and rotation of the branches connecting End Trachea to End LMB and End TRACHEA to End RMB of second tree to minimize the angle between the respective

branches in first tree. Prior knowledge of these three branchpoints in both trees is required for registration process. The rigid registration algorithm development is not a part of this thesis. It was provided by Tschirren et al. [23, 24].

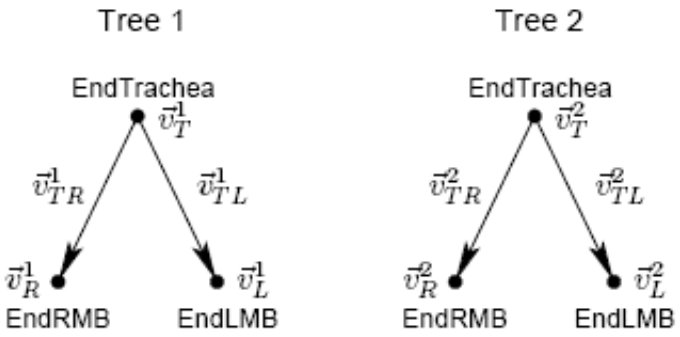



Figure 3.5: Nomenclature for rigid registration. Anatomical names correspond to names in Figure 2.3. Figure taken from [24]

Let $\psi_{yzavg}, \psi_{xzavg}, \psi_{xyavg}$ be the rotation angles around X, Y and Z axes respectively such that-

$$\psi_{yzavg} = \Bigg[\psi_{diff}\Bigg(\tan^{-1}\frac{\vec{v^1}_{TRy}}{\vec{v^1}_{TRz}}, \tan^{-1}\frac{\vec{v^2}_{TRy}}{\vec{v^2}_{TRz}}\Bigg) + \psi_{diff}\Bigg(\tan^{-1}\frac{\vec{v^1}_{TLy}}{\vec{v^1}_{TLz}}, \tan^{-1}\frac{\vec{v^2}_{TLy}}{\vec{v^2}_{TLz}}\Bigg)\Bigg]/2 \tag{3.1}$$

$$\psi_{xzavg} = \Bigg[\psi_{diff}\Bigg(\tan^{-1}\frac{\vec{v^1}_{TRx}}{\vec{v^1}_{TRz}}, \tan^{-1}\frac{\vec{v^2}_{TRx}}{\vec{v^2}_{TRz}}\Bigg) + \psi_{diff}\Bigg(\tan^{-1}\frac{\vec{v^1}_{TLx}}{\vec{v^1}_{TLz}}, \tan^{-1}\frac{\vec{v^2}_{TLx}}{\vec{v^2}_{TLz}}\Bigg)\Bigg]/2 \tag{3.2}$$

$$\psi_{xyavg} = \Bigg[\psi_{diff}\Bigg(\tan^{-1}\frac{\vec{v^1}_{TRx}}{\vec{v^1}_{TRy}}, \tan^{-1}\frac{\vec{v^2}_{TRx}}{\vec{v^2}_{TRy}}\Bigg) + \psi_{diff}\Bigg(\tan^{-1}\frac{\vec{v^1}_{TLx}}{\vec{v^1}_{TLy}}, \tan^{-1}\frac{\vec{v^2}_{TLx}}{\vec{v^2}_{TLy}}\Bigg)\Bigg]/2 \tag{3.3}$$

where $\psi_{diff}(\psi_1, \psi_2)$ is the smallest of the difference angles between $\psi_1$ and $\psi_2$.

$$\psi_{diff}(\psi_1, \psi_2) = \begin{cases} \psi_1 - \psi_2 & \text{if } |\psi_1 - \psi_2| \leqslant \pi \\ 2\pi - (\psi_1 - \psi_2) & \text{if } (\psi_1 - \psi_2) > \pi \\ 2\pi + (\psi_1 - \psi_2) & \text{if } (\psi_1 - \psi_2) < 0 \end{cases} \tag{3.4}$$

In the matrix representation, the 3D transformation can be written as-

$$[x_T, y_T, z_T]' = T \cdot [x, y, z]' \tag{3.5}$$

where $[x, y, z]$ are the original coordinates and $[x_T, y_T, z_T]$ are the transformed coordinates.

The transformation matrix T is given by -

$$\begin{bmatrix} 1 & 0 & 0 & \vec{v^1}_{Tx} \\ 0 & 1 & 0 & \vec{v^1}_{Ty} \\ 0 & 0 & 1 & \vec{v^1}_{Tz} \\ 0 & 0 & 0 & 1 \end{bmatrix} . R_z . R_x . R_y . \begin{bmatrix} 1 & 0 & 0 & \vec{v^2}_{Tx} \\ 0 & 1 & 0 & \vec{v^2}_{Ty} \\ 0 & 0 & 1 & \vec{v^2}_{Tz} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.6}$$

where

$$Rx = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\psi_{yzavg} & -\sin\psi_{yzavg} & 0 \\ 0 & \sin\psi_{yzavg} & \cos\psi_{yzavg} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$Ry = \begin{bmatrix} \cos\psi_{xzavg} & 0 & \sin\psi_{xzavg} & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\psi_{xzavg} & 0 & \cos\psi_{xzavg} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.7}$$

$$Rz = \begin{bmatrix} \cos\psi_{xyavg} & -\sin\psi_{xyavg} & 0 & 0 \\ \sin\psi_{xyavg} & \cos\psi_{xyavg} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

### 3.3.4 Matching main branchpoints

As per the definition of an association graph, it is possible to create an association graph which has a vertex corresponding to every possible pair of vertices in the two graphs to be matched. In case of human airway trees, it is common to have 100-200 vertices in each graph and such a construction of association graph would mean building a structure with as many as 10,000-40,000 vertices. Finding a maximum clique in such a large graph would be computationally expensive. To reduce the computation time we need to reduce the number of potential matches. This can be done by selectively adding vertices to the association graph based on certain constraints like topological distance, geometric features etc. Thus, vertices are added to the association graph only if certain criteria are met. The building of association graph is also performed into two steps. In this two step process, an association graph is first built for matching the main branchpoints and then after the main branchpoints are assigned via maximum clique (See Section 3.3.2), an association graph is built for every sub-tree to be matched. In this way, we split the problem of finding maximum clique into several smaller problems and hence reduce computation time and complexity. The basic idea of splitting the problem remains the same as proposed by Tschirren et al [23, 24].

#### 3.3.4.1 Adding vertices to the association graph

The main branchpoints within an airway tree are those points which are common to the airway tree structure across subjects and those that can be identified by the anatomical labels given to them. There can be up to 33 such identifiable branchpoints in a typical human airway tree (See Chap 2 Section 2.1 Figure 2.3). The

characteristic feature of these branchpoints is that they mark the beginning of sub-trees in the different lobes and sub-lobes of the lung and hence can be distinguished from the rest based on the expanse of tree beneath them. Before adding vertices to the association graph for matching main branchpoints, we first need to find and isolate only the main branchpoints. This can be done by using the branchpoint location information (spatial information). For every point in the tree, we find out the spatial extents of the sub-tree beneath it. For an edge e, let $x_{\min}$, $x_{\max}$, $y_{\min}$, $y_{\max}$, $z_{\min}$ and $z_{\max}$ be the minimum and maximum x, y and z coordinates of the sub-tree beneath it. Then the x, y and z spans are given by $spanX = x_{\max} - x_{\min}$, $spanY = y_{\max} - y_{\min}$ and $spanZ = z_{\max} - z_{\min}$ [23]. If the maximum of the x, y and z spans is greater than or equal to 25 mm (empirically determined), then the point is identified as a major branchpoint. After identifying the main branchpoints, the next step is to add the vertices to association graph. Vertices are added to the association graph only if the differences in the corresponding x, y and z spans of the two vertices to be matched are greater than or equal to 30 mm (empirically determined). This reduces the number of potential matches and hence the number of vertices added to the association graph. Using a conservative approach, we thus, restrict the size of the association graph without affecting the result of matching.

#### 3.3.4.2 Adding edges to the association graph

The addition of edges to the association graph depends on the following constraints -

1. Inheritance relationship

2. Topological distance

3. Euclidean distance

4. Angle

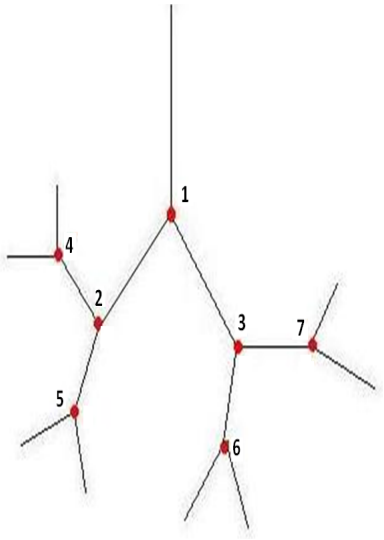
5. Path-length

Let $G = (V_{assoc}, E_{assoc})$ be the association graph for trees $T_a = (V_a, E_a)$ and $T_b = (V_b, E_b)$. Let $V_{assoc1}$ represent a match between vertices $V_{a1}$ and $V_{b1}$ and $V_{assoc2}$ represent a match between vertices $V_{a2}$ and $V_{b2}$.

**Inheritance relationship and topological distance:** The vertex relationship array $R_v$ is a required to start building an association graph [23](See Figure **??**). For a tree with $n$ nodes, $R_v$ is a two-dimensional $n \times n$ array. A cell (s,t) in $R_v$ where s = source vertex and t = target vertex contains the inheritance relationship and topological distance information between s and t where,

inheritance relationship $r(s,t) \in \{PARENT, CHILD, SIBLING, N/A\}$. For $s = t$, $r(s,t) = N/A$, topological distance $d(s,t) = 0$. For $s \neq t$, $d(s,t) >= 1$. The relationships of PARENT and CHILD are assigned to s and t respectively if t is a descendent of s (i.e $r(s,t) = PARENT, r(t,s) = CHILD$). If neither of them is a direct descendant of the other, they are siblings ($r(s,t) = r(t,s) = SIBLING$). $R_v$ is calculated by using algorithm based on a breadth-first search. (See Appendix A on page 57).

While adding vertices to the association graph, the vertices to be matched must be in the same inheritance relationship to each other and the topological distances of the two should not differ by more than +/-2 [23]. Vertices which satisfy these constraints will be further tested for possible matches. Hence $V_{assoc1}$ and $V_{assoc2}$ will be joined by an edge if and only if -




(a)

| Vertex | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 1 | N/A | PARENT | PARENT | PARENT | PARENT | PARENT | PARENT |
| 2 | CHILD | N/A | SIBLING | PARENT | PARENT | SIBLING | SIBLING |
| 3 | CHILD | SIBLING | N/A | SIBLING | SIBLING | PARENT | PARENT |
| 4 | CHILD | CHILD | SIBLING | N/A | SIBLING | SIBLING | SIBLING |
| 5 | CHILD | CHILD | SIBLING | SIBLING | N/A | SIBLING | SIBLING |
| 6 | CHILD | SIBLING | CHILD | SIBLING | SIBLING | N/A | SIBLING |
| 7 | CHILD | SIBLING | CHILD | SIBLING | SIBLING | SIBLING | N/A |

(b)

Figure 3.6: For a given tree in Figure(a), Figure(b) shows the inheritance relationships in the vertex-relationship array.

$$r(v_{a1}, v_{a2}) = r(v_{b1}, v_{b2}) \quad and \quad \left|(d(v_{a1}, v_{a2}) - d(v_{b1}, v_{b2}))\right| \quad \leqslant \quad 2 \qquad (3.8)$$

Airway trees undergo deformation with change in lung pressure. Hence, even the intra-patient scans taken at FRC and TLC show variation in the size of the tree (number of nodes detected by segmentation algorithm), spatial expanse etc. Nonetheless, the variation in spatial positions of branchpoints and hence geometric measures such a Euclidian distance between two points, path-length, etc., remain within certain limits. Although we may not be able to guarantee a match based on these features, we can use such measures to decide whether a match is likely to occur and hence reduce the problem size. Inheritance relationship and topological distance are the measures which are independent of the airway tree deformation, but the rest of the measures described below depend on airway tree deformation.

**Euclidean distance:** The Euclidean distance between any two adjacent vertices is the length of the edge joining the two vertices. For two vertices in the association graph to be eligible as a possible match, the difference in Euclidean distances for the pair of vertices in the individual trees should not be more than 20% [23]. Let $V_{a1}$, $V_{a2}$ be the two adjacent vertices in tree $T_a$ and let $V_{b1}$, $V_{b2}$ be the two adjacent vertices in tree $T_b$. Then, $V_{assoc1}$ and $V_{assoc2}$ will be joined by an edge in the association graph if and only if -

$$|(EDist(V_{a1}, V_{a2}) - EDist(V_{b1}, V_{b2})| \quad \leqslant \quad 0.2 \times EDist(V_{a1}, V_{a2}) \tag{3.9}$$

In other words,

$$\left(\frac{EDist(V_{b1}, V_{b2})}{EDist(V_{a1}, V_{a2})}\right) \geq \quad 0.8units \tag{3.10}$$

*Note that this criterion is not valid if* - $r(V_{a1}, V_{a2}) = r(V_{b1}, V_{b2}) = SIBLING$.

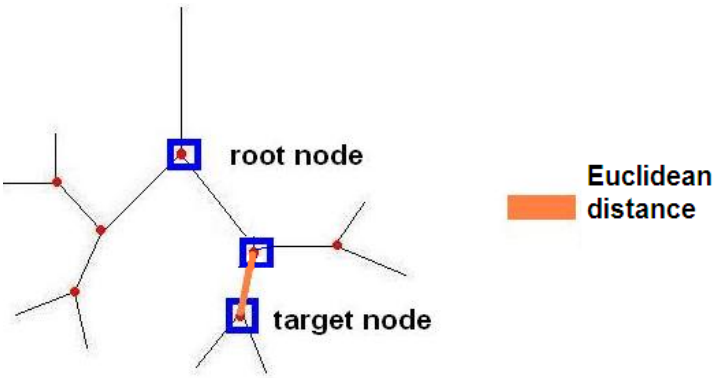



Figure 3.7: Measurement of Euclidian distance

**Angle:** Let $\vec{a_{12}}$ and $\vec{b_{12}}$ be the two vectors representing edges ($V_{a1}$,$V_{a2}$) and ($V_{b1}$,$V_{b2}$) in trees $T_a$ and $T_b$ respectively. An edge will be added to the association graph between $V_{assoc1}$ and $V_{assoc2}$ if and only if the angle $\phi$ between these two vectors is less than one radian [23].

$$\phi \quad < 1 \quad \text{radian} \tag{3.11}$$

where

$$\phi \quad = \quad \cos^{-1}\left(\frac{(\vec{a_{12}} \cdot \vec{b_{12}})}{|\vec{a_{12}}| \cdot |\vec{b_{12}}|}\right)$$

*Note that this criterion is not valid if* - $r(V_{a1}, V_{a2}) = r(V_{b1}, V_{b2}) = SIBLING$.

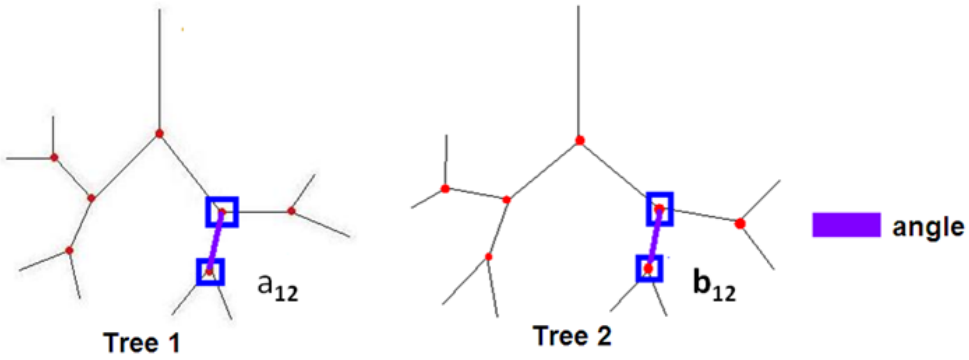



Figure 3.8: Measurement of path angle

**Path-length:** A path is any sequence of nodes $u_0u_1u_2 \ldots u_n$ such that for all $i = 1 \ldots n$, $u_{i-1}$ is adjacent to $u_i$. The length of the path at a point $u_i$ is the summation of lengths of paths for all points $u_j$, where $j = 0, 1, \ldots i$, measured from the root node which is the carina given by $u_0$, i.e., length of a path is summation of lengths of all edges lying in that path and measured from carina. Thus the path-length for any vertex $V_n$ in a tree T is given by -

$$Plength(V_n) \quad = \quad \sum_{i=0}^{n-1} \big(EDist(V_i, V_{i+1})\big) \tag{3.12}$$

Edges will be added to the association graph if and only if the difference in the normalized lengths of the two vertices to be matched is less than 10% of the maximum path-length. So an edge will be added to the association graph between $V_{assoc1}$ and $V_{assoc2}$ if and only if -

$$Plength(V_{a1}) - Plength(V_{b1}) < 0.1 \times (\max(\max(PlengthT_1), \max(PlengthT_2))$$

and

$$Plength(V_{a2}) - Plength(V_{b2}) < 0.1 \times (\max(\max(PlengthT_1), \max(PlengthT_2)) \tag{3.13}$$

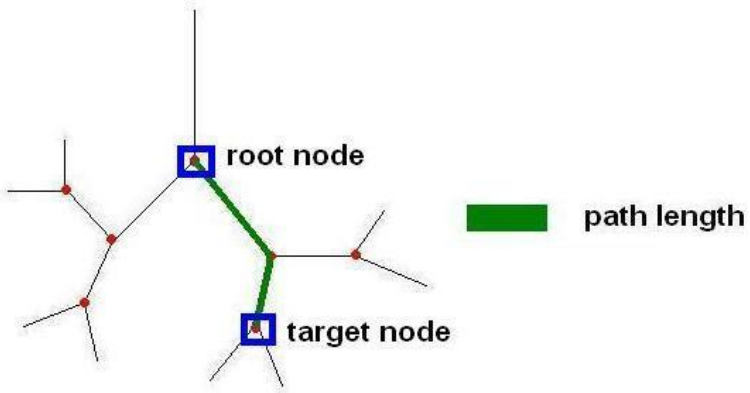



Figure 3.9: Measurement of path length

### 3.3.5 Matching sub-trees

#### 3.3.5.1 Adding vertices to association graph of sub-tree

After matching the main branchpoints, the sub-trees underneath them will be matched. For every pair of matched main branchpoints, all the branchpoints lying within a distance $d_{max} = 60$ mm below them will be considered for matching (empirically determined). The smaller the value of $d_{max}$, the lesser will be the number of vertices added, reducing the size of the association graph and the maximum clique and thus, reducing the computation time. This means that there exists a possibility that the maximum clique for a larger association graph considering more points in the neighborhood will be different. On the other hand, having too many points in the association graph will tend to increase the size of the maximum clique and hence the computation time. We thus set this value such that the size of the association graph is neither too large nor too small.

#### 3.3.5.2 Adding edges to association graph of sub-tree

The constraints for adding edges to association graph for sub-trees are the same as those for matching main branchpoints.

*See section 3.3.4.2 on page 22.*

### 3.3.6 Finding the maximum clique

A number of heuristic (approximate) as well as exact algorithms are available for solving the maximum clique problem. The heuristic methods may not give the exact optimal solution but the computation time required is less as compared to exact algorithms. Such approaches are useful when it is important to keep the computation time low. The exact algorithms are computationally expensive but guarantee an

optimal solution [23]. Hence there is a trade-off between how optimal a solution we seek and the time required. In our case, we want an optimal solution while keeping the time within limits. An exact algorithm can be applied to our problem, provided the problem size is be reduced by dividing the problem into smaller parts. As seen in the previous steps, the problem size and effectively the size of the association graph has been reduced by matching the sub-trees instead of the entire tree. Hence we can use an exact matching algorithm.

*The algorithm for finding the maximum non-weighted clique can be found in Appendix A on page 57.*

### 3.3.7 Implementation

The development of segmentation and skeletonisation algorithms is not a part of this thesis. They have been used as is from Pulmonary Workstation [2]. The matching part of the algorithm has been developed in C++ and all the graph-related operations have been implemented using the BOOST Graph Libraries [21]. The matching algorithm has been implemented as a command-line application. It takes two XML tree files as input and outputs another tree-matching XML file. The ground truth data establishment has been done by the author and has been supervised and checked by a human expert, Dr. Geoffrey McLennan. The validation has been done using an interactive program (developed by Martin Urschler) as illustrated in Figure 3.10 which allows matching of the two trees by visual inspection.


Figure 3.10: Interactive program for hand-matching airway trees.

# CHAPTER 4
# RESULTS

While doing a review of literature, we identified some of the shortcomings of the previous approach [23] [24]. Since it is label-dependent, incorrect labeling of a branchpoint can lead to incorrect matching. Such errors are also likely to propagate to the sub-tree beneath the incorrectly matched branchpoints. Furthermore, failure to label a branchpoint in either tree means finding no match for that branchpoint and implies that the sub-tree underneath it would not be discovered either. Upon testing the algorithm on normal human cases, we found that the labeling errors for trifurcation points reduced the accuracy of the results (accuracy calculated without accounting for the branchpoints missed (i.e. not matched) by algorithm). It was observed that the number of matches present in ground truth were significantly more than the number of matches produced by the method i.e., the number of missing matches in the result was significant. Hence, the labeling errors or failure to label branchpoints was responsible for the low number of matches. The figure 4.1 shows a label-wise analysis of missed and mismatched main branchpoints tested on seven normal human cases.

It can be seen that the incidence of missing the prominent labels like TriLUL can be quite high.

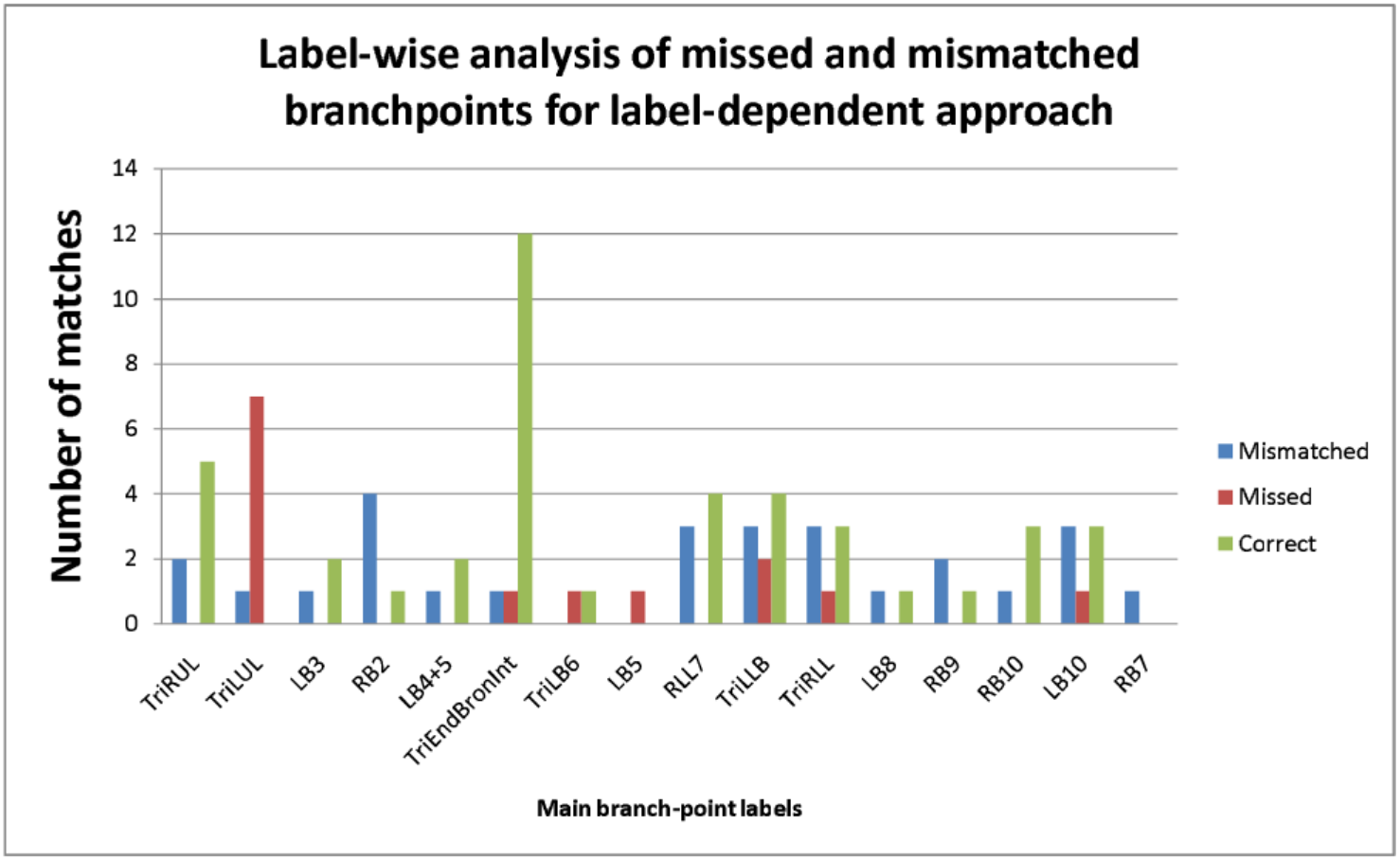



Figure 4.1: Label-wise analysis of incorrect and correct main branchpoint matches generated by [23]

The detailed analysis of the results for the seven normal cases is as shown in the table 4.1. Here, we calculate the incorrect matches not only in the conventional way where *Incorrect = Mismatched* but also in a novel way where we also consider those matches that were missed by algorithm hence making *Incorrect = Mismatched + Missed*. It can be seen that if we account for the missed matches, then accuracy of the algorithm decreases.

In the table 4.1, we define the following: Total Ref Matches = total number of matches present in ground truth or reference (considered as total correct),
Verified matches = matches produced by the algorithm verified with the matches present in reference (classified as Correct and Incorrect),
Comp Matches Not In Ref = matches produced by the algorithm but absent in reference (considered as mismatch) and
Ref Matches Not In Comp = matches present in the reference but absent in those produced by the algorithm (considered as missed)

Our proposed change to the algorithm not only improves the matching accuracy but also matches more number of branchpoints per case. We tested our algorithm on 21 normal and 6 diseased cases. The results achieved by making matching independent of labeling not only improve the matching accuracy (mismatches are corrected) but also increase the total number of correct matches (missed matches are discovered). To compare the results obtained by our algorithm with the previously proposed one, we use the same 27 cases for testing both the algorithms. The figures 4.2, 4.3, 4.4 and 4.5 show the analysis of 21 normal cases and 6 diseased cases when matched using the label-dependent and label-independent approaches.

| Subject | Total Ref Matches | Verified Matches | | Comp Matches Not In Ref | Ref Matches Not In Comp | Incorrect = Missed + Mismatched | | Incorrect = Mismatched | |
|---|---|---|---|---|---|---|---|---|---|
| | | Correct | Incorrect | | | %Incorrect | %Correct | %Incorrect | %Correct |
| 1 | 17 | 11 | 0 | 1 | 6 | 41.18 | 58.82 | 0 | 100 |
| 2 | 21 | 16 | 4 | 0 | 1 | 23.81 | 76.19 | 19.05 | 80.95 |
| 3 | 20 | 11 | 6 | 0 | 3 | 45.00 | 55.00 | 30.00 | 70.00 |
| 4 | 20 | 12 | 7 | 1 | 1 | 45.00 | 55.00 | 35.00 | 65.00 |
| 5 | 20 | 15 | 5 | 0 | 0 | 25.00 | 75.00 | 25.00 | 75.00 |
| 6 | 26 | 14 | 8 | 1 | 4 | 50.00 | 50.00 | 30.77 | 69.23 |
| 7 | 24 | 12 | 3 | 0 | 9 | 50.00 | 50.00 | 12.50 | 87.5 |
| avg | 21.14 | 13 | 4.71 | 0.43 | 3.43 | 40.00 | 60.00 | 21.76 | 78.24 |

Table 4.1: Case-wise analysis of matches generated by method of Tschirren et al. [23] [24]

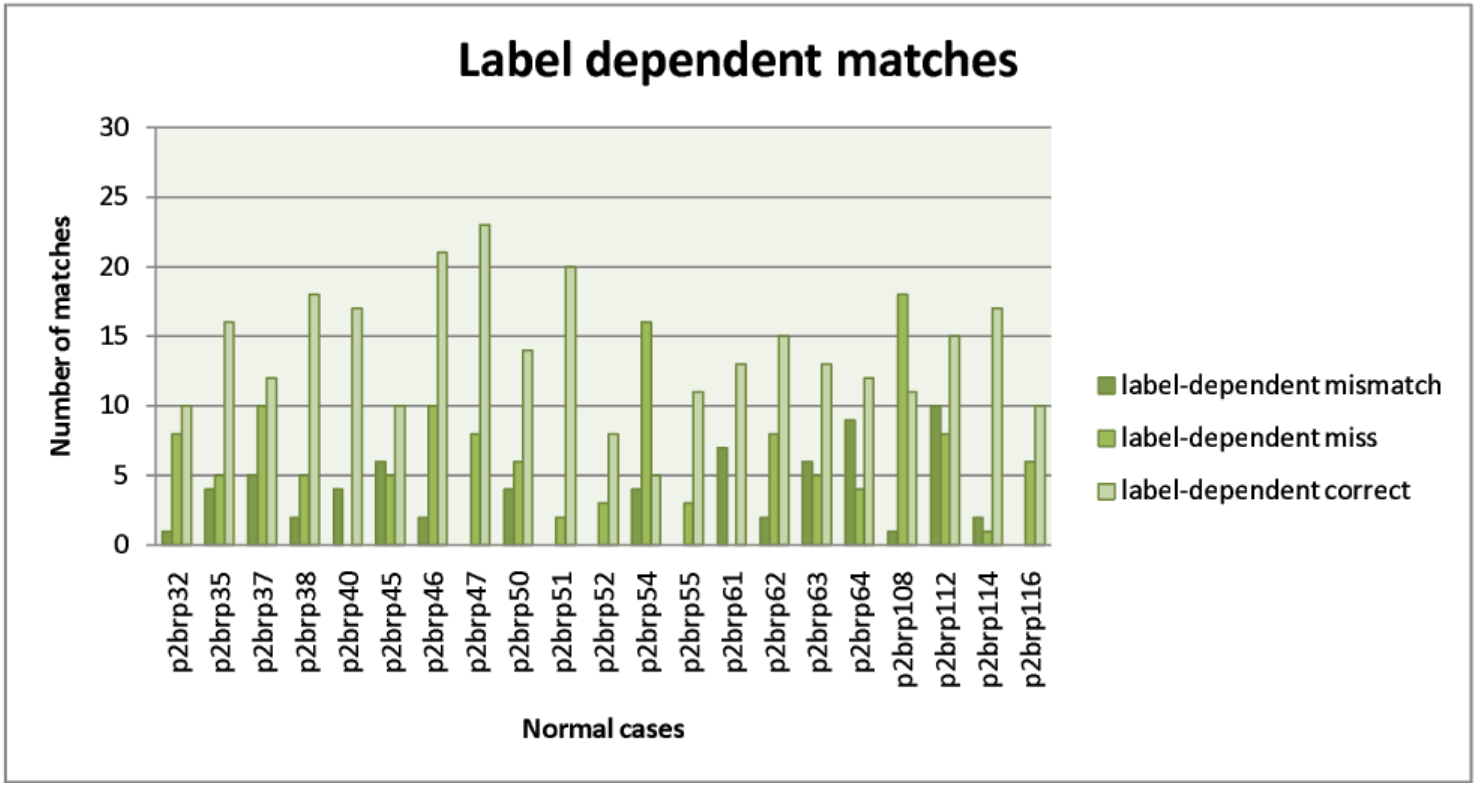



Figure 4.2: Normal Cases: Case-wise analysis of matches using label-dependent approach [23].

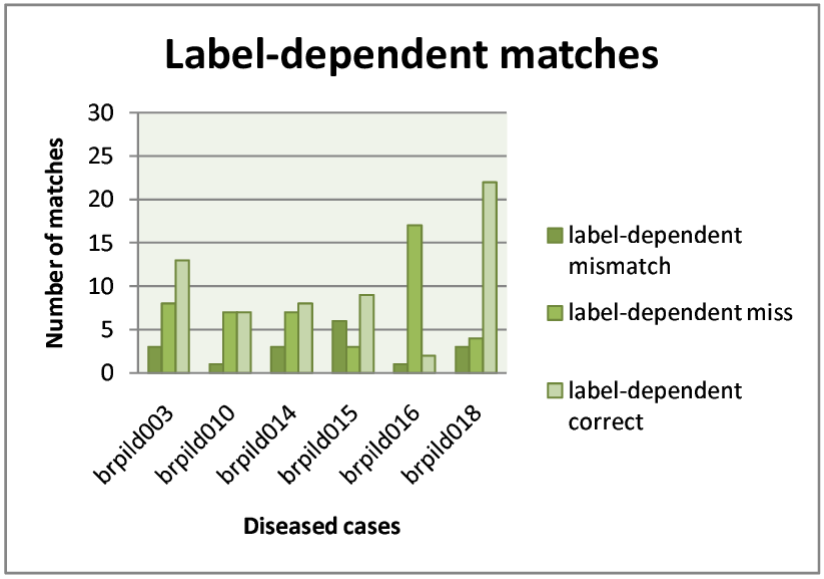



Figure 4.3: Diseased Cases: Case-wise analysis of matches using label-dependent approach [23].

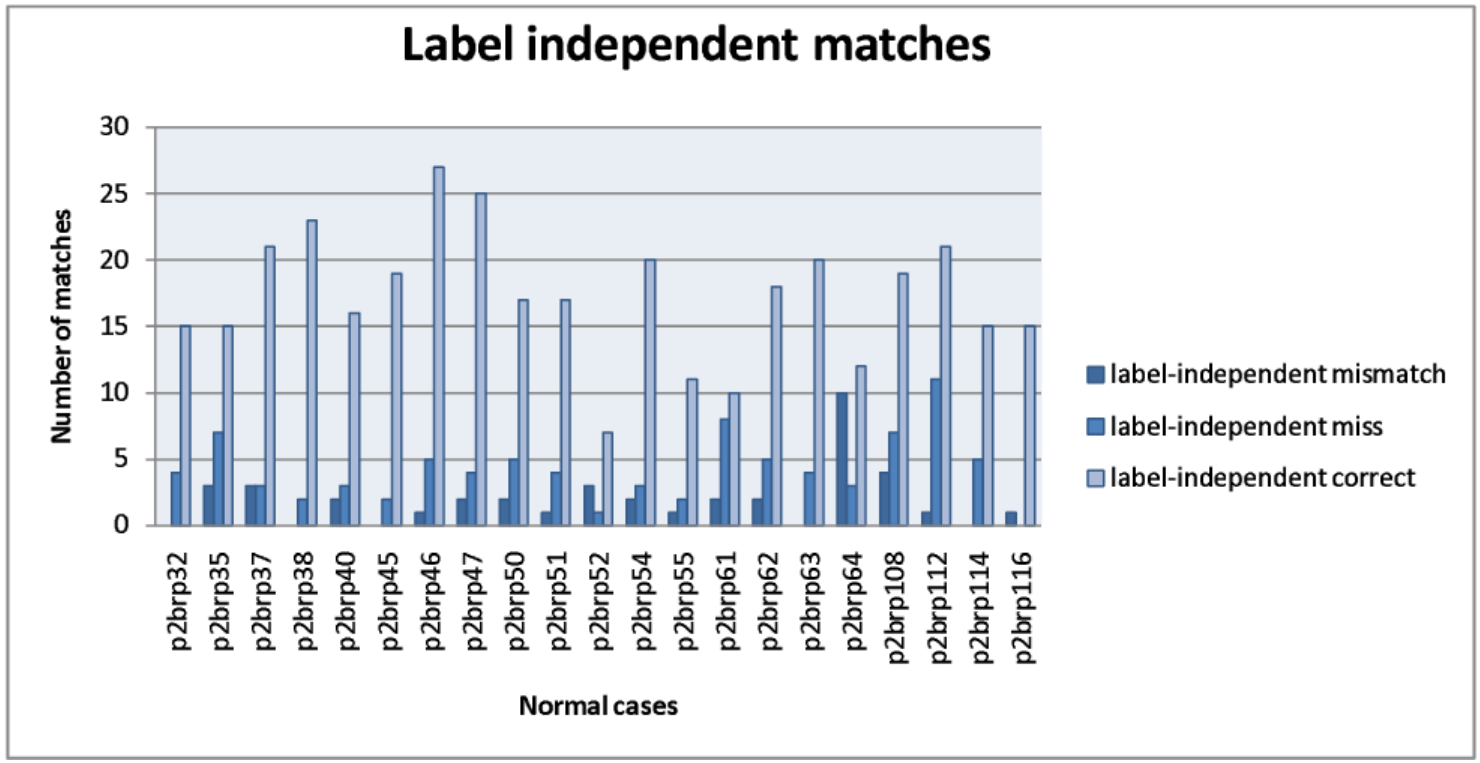



Figure 4.4: Normal Cases: Case-wise analysis of matches using label-independent approach.

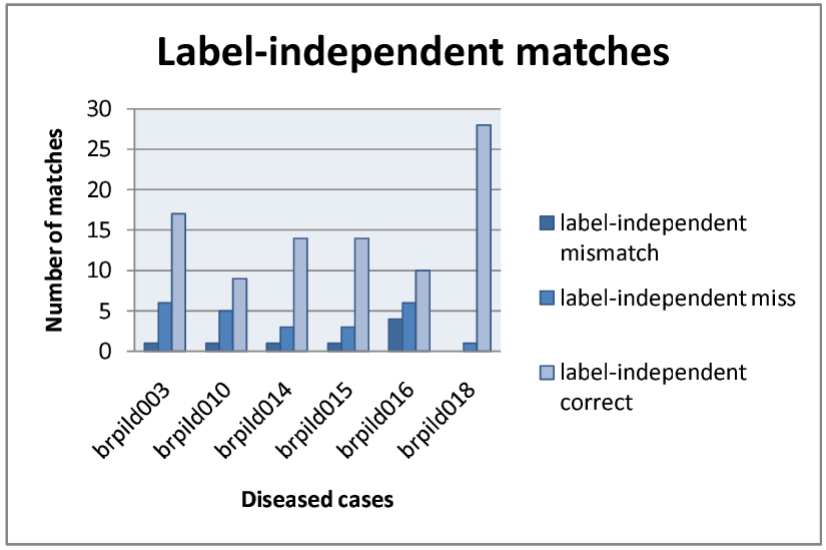



Figure 4.5: Diseased Cases: Case-wise analysis of matches using label-independent approach.

From the figures 4.2 and 4.3 it can be observed that out of the total number of branchpoint matches, a significant number is either missed or mismatched. However, making the approach label-independent reduces the number of incorrect matches as can be seen in the figures 4.4 and 4.5.

The figures 4.6 and 4.7 show a comparison of both approaches on the same normal and diseased cases.

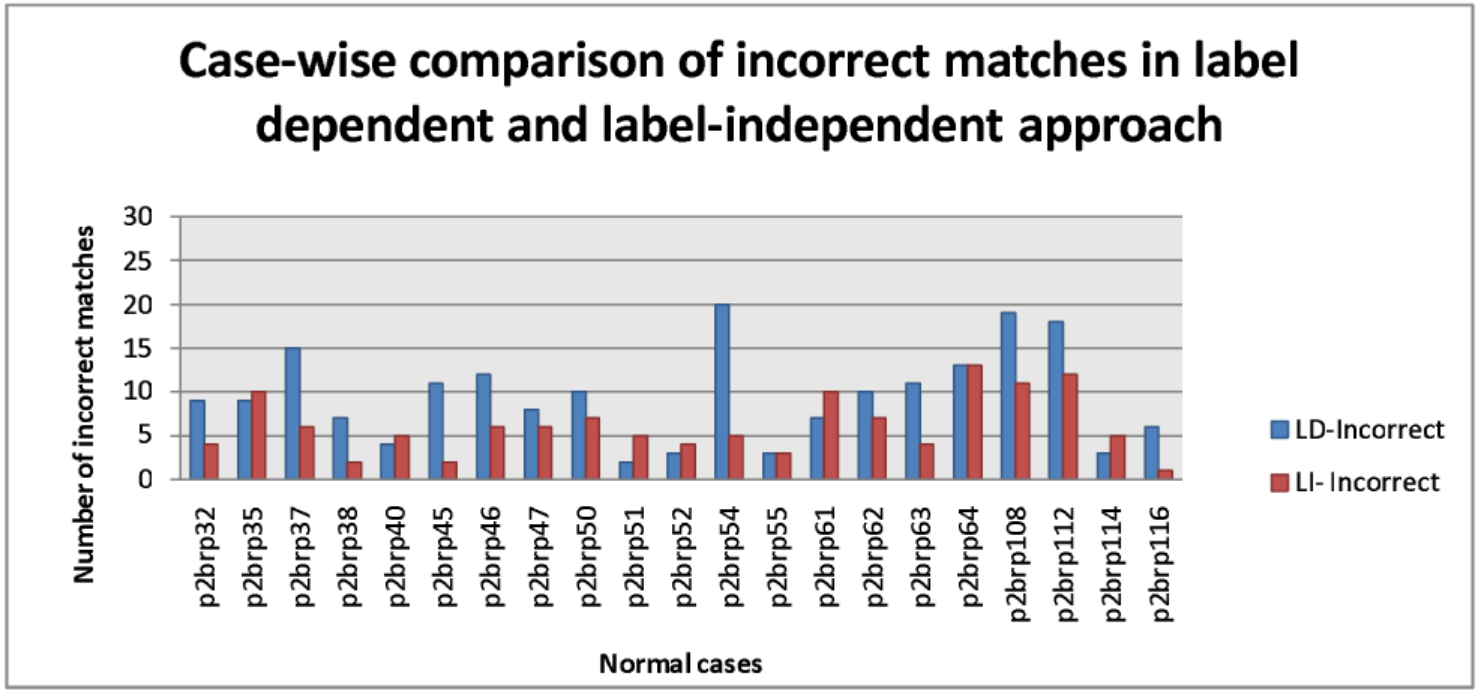



Figure 4.6: Normal Cases: Case-wise comparison of incorrect matches between label-dependent and label-independent approach

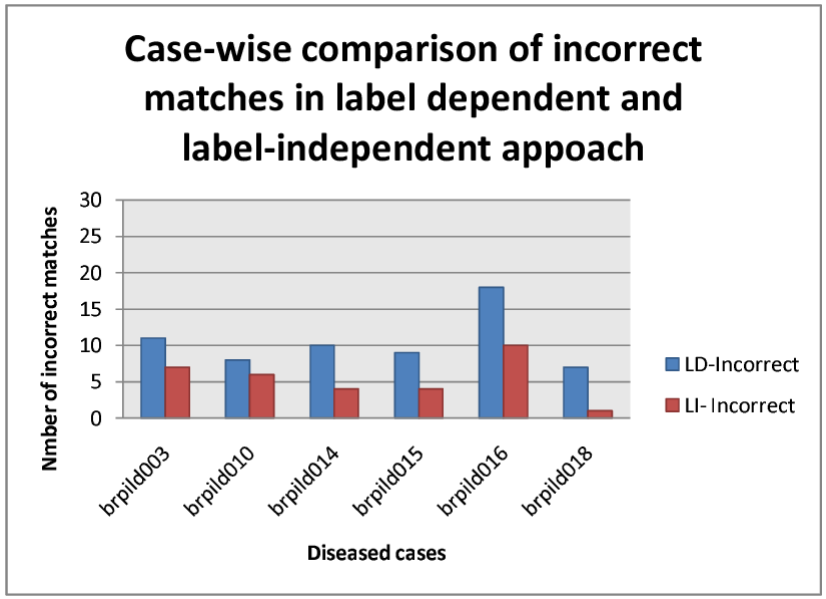



Figure 4.7: Diseased Cases: Case-wise comparison of incorrect matches between label-dependent and label-independent approach

From the figures above, it can be seen that out of the 21 normal and six diseased cases, at least 13 normal and all diseased cases exhibit more number of incorrect matches when using the label-dependent approach. The average error for the label-dependent approach in these cases is 46.8% compared to an average error of 20.89% while using the label-independent approach. Thus, there are 25.91% more incorrect matches when matching is done via the label-dependent approach as compared to label-independent approach. In the remaining eight normal cases, six exhibit more incorrect matches using label-independent approach whereas two have same number of incorrect matches for both approaches. The average error in these cases while using the label-independent approach is 32.98% compared to 23.57% while using the label-dependent approach. However, there are only 9.41% more incorrect matches when matching is done via the label-independent approach as compared to label-dependent approach.

The percentage accuracy when calculated for the 27 cases is summarized as shown in Table 4.2.

| Percentage average accuracy | | | | |
|---|---|---|---|---|
| | Label Dependent | | Label Independent | |
| | Normal | Diseased | Normal | Diseased |
| Incorrect=missed+mismatched | 59.27 | 49.19 | 73.93 | 74.19 |
| Incorrect=mismatched | 85.95 | 86.29 | 91.85 | 93.54 |

Table 4.2: Percentage average accuracy

Here,

$$Percentage \quad accuracy = \left[\frac{Total \quad matches - Incorrect \quad matches}{Total \quad matches}\right] \times 100 \quad (4.1)$$

Thus, we achieve an overall matching accuracy of

73.98% if *Incorrect = Missed + Mismatched*

and

92.19% if *Incorrect = Mismatched*.

The average improvement in accuracy for normal and diseased cases is 14.66% and 25.0% respectively whereas the average percent increase in the total number of correctly matched branch points for normal and diseased cases is 24.74% and 50.81% respectively.

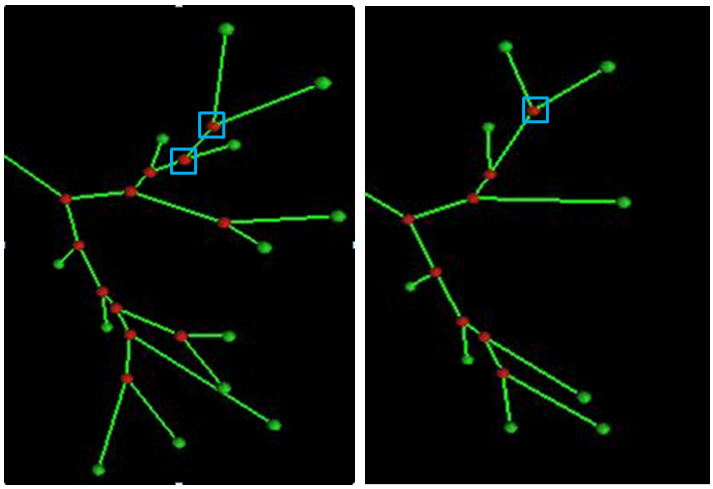
However the average time taken for matching in normal cases is 21.09 sec (range: 0.01- 211s) and for diseased cases is 45.67 sec (range: 0.01- 262.56s) which is more than the earlier approach which is close to 1-2 sec. More details will be discussed in Chap 5 on page 44.

Thus we present an improved association graph approach which not only improves the matching accuracy but also increases the total number of correct matches.

# CHAPTER 5
# DISCUSSION

In this section, we would like to address a few issues that arise in the tree-matching and explain either how they are handled by the algorithm or their impact on the algorithm.

Firstly, matching of trees involves the matching of corresponding branchpoints within the two trees. In a graph representation of a tree, these branchpoints will be modeled as nodes and the segments will be modeled as edges. Human airway trees are mostly binary trees (with few trifurcations) with every node except the terminal node branching into two branches. As a result, there is redundancy in the branching pattern. For intra-patient trees, this redundancy is more evident. This can sometimes lead to ambiguities while matching the trees. One such example is shown below.



(a) (b)

Figure 5.1: Sometimes there can be ambiguity in matching corresponding branch-points in two trees as shown in Figure(a)(FRC scan) and (b)(TLC scan). For the given branchpoint (marked with square) in Figure(b), there seem to be two possible matches in Figure(a). (marked by squares).

While matching such trees, the local information as well as the global information is utilized to find the best match. The local information using constraints such as the inheritance relationship, topological distance, euclidian distance, path-length, angle are instrumental in deciding whether the ambiguous point qualifies as a match or not. Once added to the association graph, the maximum clique in the graph, which is a more global measure (since finding maximum clique is finding the maximum number of connected vertices) will decide the correct match depending on addition of what pair yields the maximum number of connected vertices.

The number of branches detected in an airway tree depends on several parameters like the quality of the chest CT scan, the performance of the segmentation and skeletonisation algorithms, artifacts etc. As a result, the segmentation result may or may not contain all the branches (e.g. the smaller terminal branches) and/or may contain false branches. Although we can, under most conditions, identify and prune the false branches, we cannot guarantee a good segmentation result in case of diseased subjects. Since our algorithm depends on the result of segmentation and skeletonisation, the quality of this result has an effect on the results of our matching algorithm.

We have tested our method with trees containing as few as 30 and as many as 286 branch points in one tree and the time required to match such trees (with another containing 105 and 78 branch points respectively) is 0.01 sec and 13.35 sec respectively.

There are several factors affecting the total time required for matching. The size of the two trees to be matched (i.e. the total number of branchpoints present in two trees), the number of main branchpoints present in two trees, the similarity

between the two trees (which determines the number of vertices and edges added to the association graph which, in turn, determine the time required to find the maximum clique), etc. all influence the time taken to match trees. These factors are not independent and hence, the exact relation between these variables and the run time for matching is unknown. However, since time required to solve the NP-complete clique problem greatly increases with increase in the problem size, the author expects the runtime to increase with an increase in the tree size; depending on how similar the trees are and how many main branchpoints are present in the two trees. It is worth noting that the time required to find the main branchpoints and the subtrees and the time required to add edges and vertices to the association graph is negligible as compared to the time required for finding the maximum clique. In other words, the time required for finding the maximum clique dominates in the contribution to total runtime. The figures 5.2, 5.3, 5.4 and 5.5 show the relation between the different parameters described above and time.

A normal human chest CT scan segmented with a good segmentation algorithm can have as many as 200-400 branch points. Matching trees with such high number of branch points by using our method may be computationally expensive. However, since matching of trees is of more significance to the diseased subjects, we expect to have fewer number of branch points in the segmentation result and hence expect to achieve matching within a reasonable time.

Among the underlying conditions required for our input data, we assume that the body orientation of the patient remains the same during both the scans, there is little or no change in tree topology and that the three labels viz. Carina (End Trachea), End RMB and END LMB are known in both trees. Also, this method has been

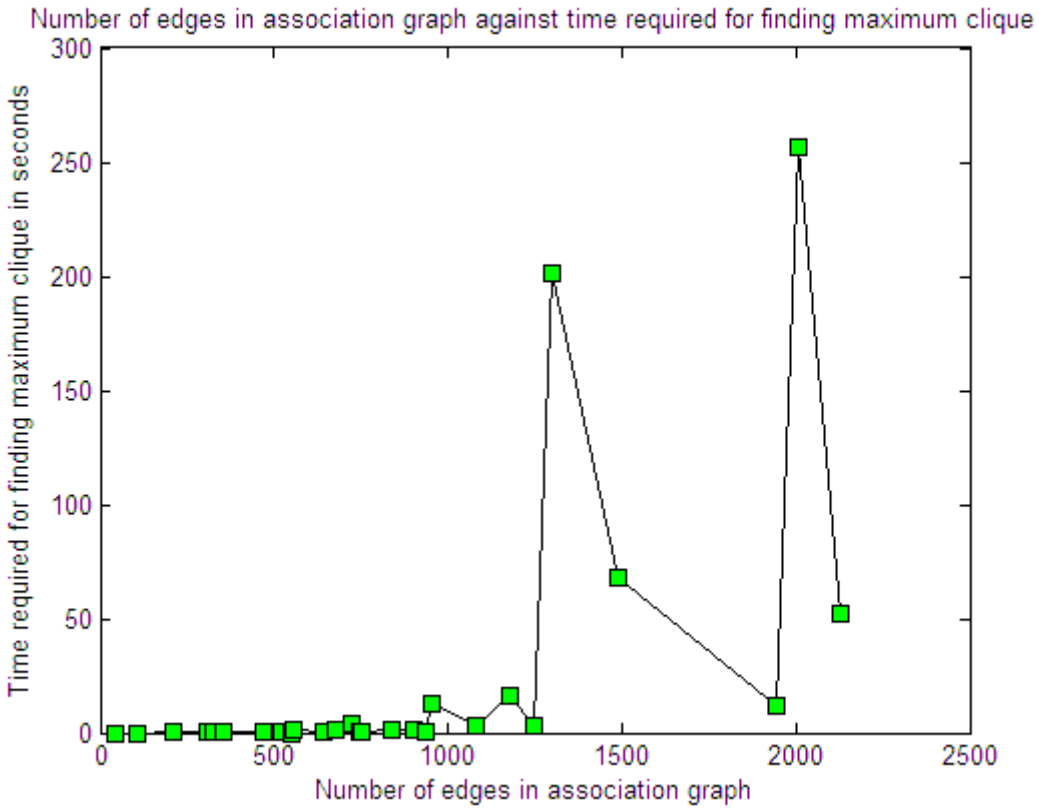



Figure 5.2: Plot of the number of edges in the association graph against time required(in sec) to find maximum clique in the association graph.

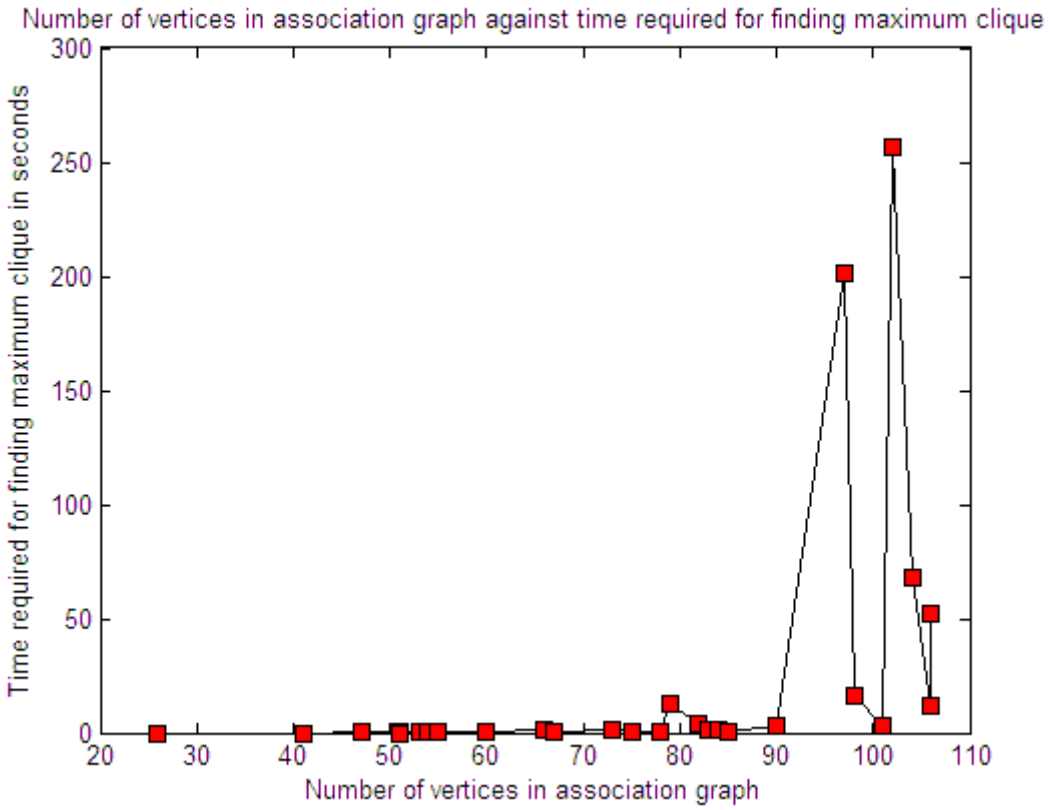



Figure 5.3: Plot of the number of vertices in the association graph against time required(in sec) to find maximum clique in the association graph.

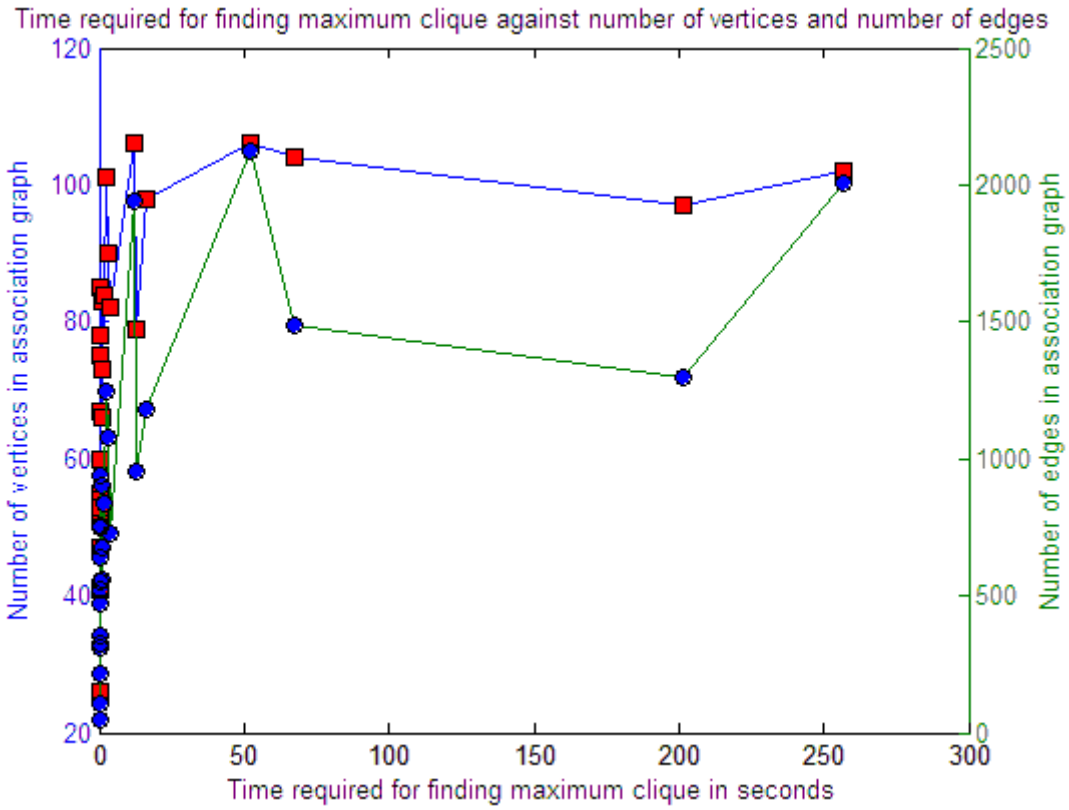



Figure 5.4: Combined plot of the number of edges and vertices in the association graph and number of vertices in the association graph gainst time required(in sec) to find maximum clique in the association graph.

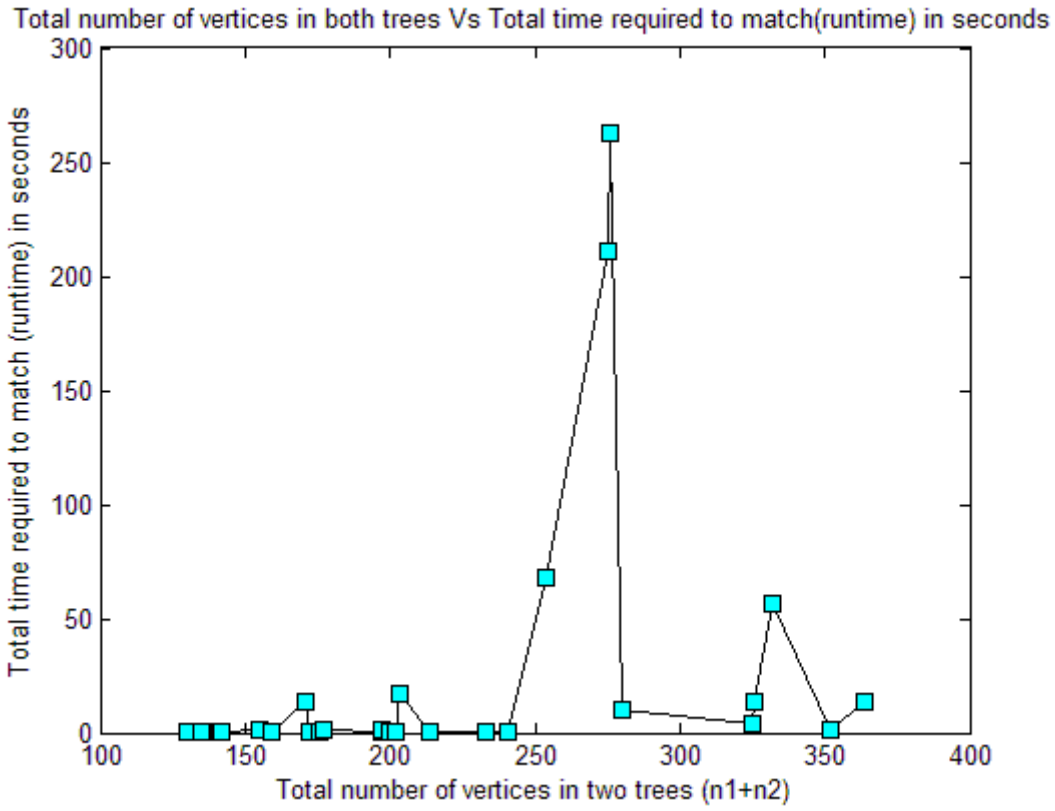



Figure 5.5: Plot of the total number of vertices in two trees against time required(in sec) for matching.

tested for intra-patient matching only and the authors have no information whether this can be used "as is" for inter-patient matching. In case of inter-patient matching, the assumptions of same body orientation, etc. may be different. There also might be a considerable difference in tree topology (for e.g. tree topology for a child and for an adult could be different, etc.). Since the airway tree structure shows a certain similarity upto the main branchpoints across subjects, these main branchpoints may be identified and matched. However, since the branching pattern becomes more random beyond these points, the matching of sub-trees using the same constraints as those for intra-patient matching, may or may not guarantee good results.

Our approach works efficiently in matching the intra-patient airway trees and is better than most of the previously proposed approaches. The results obtained by the different approaches are summarized in the Table 5.1.

While calculating accuracy, it is important to address the issue of sensitivity and specificity. Statistically, the mismatches would be classified as "False Positives" and the missed matches as "False Negatives". Since in our approach it is not possible to define "True Negatives", specificity, which measures the proportion of negatives which are correctly identified, is 0%. Our aim, however, is to achieve higher sensitivity which measures the proportion of actual positives which are correctly identified. In our approach, we have redefined the way incorrect matches are calculated (incorrect = missed + mismatched). Hence the sensitivity of our approach is 79.71% while accuracy is 73.98%.

| **Author** | **Method** | **Organ** | **No.of datasets** | **Percent Accuracy** | **Run-time(sec)** |
|---|---|---|---|---|---|
| Pisupati et al. [20] | Central axis matching | lung | 1 | unknown | unknown |
| Tschirren et al. [23, 24] | Association graph | lung | 17 | 92.90 | 1 to 3 |
| Charnoz et al. [7] | Tree search | liver: real | 1 | 90 | 240 |
| | | liver: virtual | 1 | 95 | unknown |
| Graham et al. [9, 8] | Dynamic programming | lung,heart | 1 | unknown | 5 |
| Kaftan et al. [10] | Path matching | lung | 10 | 87 | unknown |
| Bulow et al. [5] | Shape features: shape context | lung | 6 | 40 | unknown |
| | Shape features: statistical moment | lung | 6 | 69 | unknown |
| Tang et al. [22] | Tree edit distance | brain | 2 | unknown | unknown |
| Metzen et al. [12] | Association graph | lung | 1 | 100 | 207.35 |
| | | liver | 1 | 84 | 369.23 |
| Lohe et al. [11] | Tree search | liver | 7 | 80.9 | 1 to 45 |
| | | lung | 4 | 89.9 | 1 to 45 |
| Bodas | Association graph | lung | 27 | 92.19 | 0.01 to 262.56 |

Table 5.1: Comparison of tree matching approaches where accuracy is calculated as *Incorrect matches = mismatched matches only.*

The matching constraints were empirically decided using 10 training datasets. The accuracy of the approach excluding these datasets was found to be 73.70% which is comparable to the overall accuracy calculated using these datasets.

We also tested both the approaches with the same 27 cases and the label-wise comparison of incorrect matches for both the approaches is as shown in figure 5.6.

The label-wise accuracy of the label-dependent approach is as shown in the table 5.2.

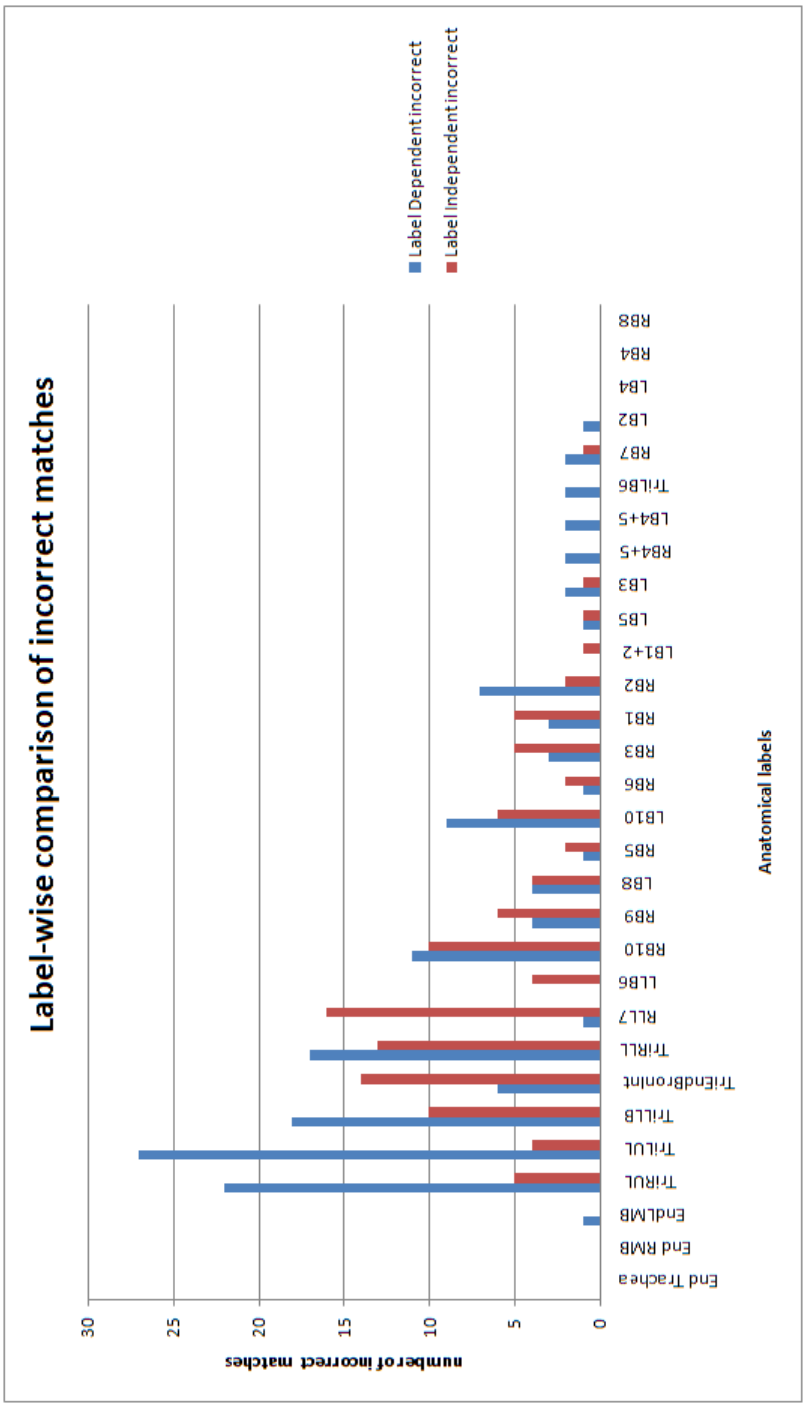



Figure 5.6: Label-wise comparison of incorrect matches for label-dependent label-independent approach.

| Label | Percent Accuracy |
|---|---|
| End Trachea | 100.00 |
| End RMB | 100.00 |
| End RMB | 100.00 |
| TriRUL | 86.84 |
| TriLUL | 86.2 |
| TriLLB | 20.58 |
| TriEndBronInt | 71.42 |
| LLB6 | 84.00 |
| TriLB6 | 100.00 |
| RB1 | 44.44 |
| RB2 | 88.23 |
| RB3 | 75.00 |
| LB1+2 | 80.00 |
| LB3 | 85.71 |
| LB4+5 | 100.00 |
| RB4+5 | 100.00 |
| RLL7 | 27.27 |
| TriRLL | 55.17 |
| LB2 | 100.00 |
| LB4 | 100.00 |
| LB5 | 50.00 |
| RB5 | 71.42 |
| RB6 | 80.00 |
| RB7 | 50.00 |
| RB9 | 25.00 |
| RB10 | 23.07 |
| LB8 | 55.55 |
| LB10 | 57.14 |
| RB4 | 100.00 |
| RB8 | 100.00 |

Table 5.2: Label-wise accuracy of label dependent approach

# CHAPTER 6
# CONCLUSIONS AND FUTURE WORK

We have introduced an improved association graph approach for matching of intra-patient airway trees. Our method is efficient in matching airway trees within a reasonable time and has a matching accuracy of 73.98% and sensitivity of 79.71%. It is important to note that we consider false positives and false negatives while calculating accuracy. This information, however, is not available for the results of earlier approaches.

The author has not yet established a mathematical relation between the number of branch points and the time taken to match those. In the future, establishing such a relation would be useful.

It may be possible to use this matching algorithm for the tree labeling discussed in [23] [24] where the reference tree is matched with the input tree to yield a labeled tree. However, the author has not implemented it.

Currently, our algorithm requires three main labels viz. Carina (End Trachea), End LMB and END RMB to begin matching. However, the dependence on these labels can be reduced and the matching may be made entirely independent of labels.

# APPENDIX A
# ALGORITHMS

## I. Algorithm for computing vertex-relationship array [23, 24]

**Input:** $\mathcal{T}$: tree
**Output:** $\mathcal{R}_v$: vertex relationship array

re-size($\mathcal{R}_v, N, N$)
**for** *every element in* $\mathcal{R}_v$ **do**
  $r \leftarrow$ "N/A"
  $d_{\min} \leftarrow d_{\max} \leftarrow 0$
breadth-first-search($\mathcal{T}$)

**function** breadthFirst($t$)
  Do breadth-first search in $t$, starting from root of $t$, and call examineEdge() for every new edge discovered.
  **return**

**function** examineEdge($e$)
  $s \leftarrow$ sourceVertex($e$)
  $t \leftarrow$ targetVertex($e$)
  **for** $x = 1 \ldots N$ **do**
    **if** $\mathcal{R}(s,x).r \neq$ *"N/A"* **then**
      **switch** $\mathcal{R}(s,x).r$ **do**
        **case** *"PARENT" or "SIBLING"*
          $\mathcal{R}(x,t).r \leftarrow \mathcal{R}(t,x).r \leftarrow$ "SIBLING"
        **case** *"CHILD"*
          $\mathcal{R}(x,t).r \leftarrow$ *"PARENT"*
          $\mathcal{R}(t,x).r \leftarrow$ *"CHILD"*
      minTopoDist $\leftarrow \mathcal{R}(s,x).d_{\min}$
      maxTopoDist $\leftarrow \mathcal{R}(s,x).d_{\max}$
      **if** $\mathcal{R}(x,t).d_{\min} = 0$ **then**
        $\mathcal{R}(x,t).d_{\min} \leftarrow$ minTopoDist
      $\mathcal{R}(x,t).d_{\min} \leftarrow \min(\mathcal{R}(x,t).d_{\min}, \text{minTopoDist})$
      $\mathcal{R}(x,t).d_{\max} \leftarrow \max(\mathcal{R}(x,t).d_{\max}, \text{maxTopoDist})$
      $\mathcal{R}(t,x).d_{\min} \leftarrow \mathcal{R}(x,t).d_{\min}$
      $\mathcal{R}(t,x).d_{\max} \leftarrow \mathcal{R}(x,t).d_{\max}$
  $\mathcal{R}(s,t).r \leftarrow$ "PARENT"
  $\mathcal{R}(t,s).r \leftarrow$ "CHILD"
  $\mathcal{R}(s,t).d_{\min} \leftarrow \mathcal{R}(t,s).d_{\min} \leftarrow \mathcal{R}(t,s).d \leftarrow \mathcal{R}(s,t).d \leftarrow 1$
  **return**

**II. Algorithm for computing maximum non-weighted clique** [23, 24] [14, 15]

```
Input: G: Graph with set of vertices V and set of edges E.
Output: C: Maximum unweighted clique.
1  max ← 0
2  clique ← ∅
3  tmp_clique ← ∅
4  _maxClique(V, 0)
5  return clique
6  function _maxClique(U, size)
7      if |U| = 0 then
8          if (size > max) then
9              max ← size
10             clique ← tmp_clique
11         return
12     while U ≠ ∅ do
13         if size +|U| ≤ max then
14             return
15         i ← min{j | v_j ∈ U}
16         U ← U \ {v_i}
17         _maxClique(U ∩ neighbors(v_i), size +1)
18     return
```

# REFERENCES


[1] Lung disease data 2008, 2008.

[2] Vida diagnostics, pulmonary workstation, 2008.

[3] E. A. Boydan. *Segmental anatomy of the lungs*. McGraw-Hill, 1955.

[4] C. M. Brown and D.H. Ballard. *Computer Vision*. Englewood Cliffs, N.J. : Prentice-Hall, 1982.

[5] Thomas Bulow, Cristian Lorenz, Rafael Wiemker, and Janett Honko. Point based methods for automatic bronchial tree matching and labeling. *Medical Imaging 2006: Physiology, Function, and Structure from Medical Images*, 6143(1):61430O, March 2, 2006 2006.

[6] Arnaud Charnoz, Vincent Agnus, Grgoire Malandain, Stphane Nicolau, Mohamed Tajine, and Luc Soler. Design of robust vascular tree matching: Validation on liver. *Information Processing in Medical Imaging*, pages 443–455, 2005.

[7] Arnaud Charnoz, Vincent Agnus, Grgoire Malandain, Luc Soler, and Mohamed Tajine. Tree matching applied to vascular system. *Graph-Based Representations in Pattern Recognition*, pages 183–192, 2005.

[8] M. W. Graham. Optimal graph-theoretic approach to 3-d anatomical tree matching. *Biomedical Imaging: Nano to Macro, 2006. 3rd IEEE International Symposium on*, pages 109–112, 2006.

[9] Michael W. Graham and William E. Higgins. Globally optimal model-based matching of anatomical trees. *Medical Imaging 2006: Image Processing*, 6144(1):614415, March 2, 2006 2006.

[10] Jens N. Kaftan, Atilla P. Kiraly, David P. Naidich, and Carol L. Novak. A novel multipurpose tree and path matching algorithm with application to airway trees. *Medical Imaging 2006: Physiology, Function, and Structure from Medical Images*, 6143(1):61430N, March 2, 2006 2006.

[11] Tobias Lohe, Tim Krger, Stephan Zidowitz, Heinz-Otto Peitgen, and Xiaoyi Jiang. Hierarchical matching of anatomical trees for medical image registration. *Medical Biometrics*, pages 224–231, 2008.

[12] Jan Metzen, Tim Krger, Andrea Schenk, Stephan Zidowitz, Heinz-Otto Peitgen, and Xiaoyi Jiang. Matching of tree structures for registration of medical images. *Graph-Based Representations in Pattern Recognition*, pages 13–24, 2007.

[13] Keith Leon Moore. *Essential Clinical Anatomy.* Lippincott Williams and Wilkins, 1995.

[14] Panos M. Pardalos, Jonas Rappe, Mauricio, and G. C. Resende. An exact parallel algorithm for the maximum clique problem. In *In High Performance and Software in Nonlinear Optimization*, pages 279–300. Kluwer Academic Publishers, 1997.

[15] Panos M. Pardalos and Jue Xue. The maximum clique problem. *Journal of Global Optimization*, 4(3):301–328, 04/01 1994. M3: 10.1007/BF01098364.

[16] Y. Park. Registration of linear structures in 3-d medical images, 2002.

[17] M. Pelillo. Matching free trees, maximal cliques, and monotone game dynamics. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(11):1535–1541, 2002.

[18] Marcello Pelillo, Kaleem Siddiqi, and Steven W. Zucker. Matching hierarchical structures using association graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11):1105–1120, 1999.

[19] C. Pisupati, L. Wolff, W. Mitzner, and E. Zerhouni. Tracking 3-d pulmonary tree structures. *Mathematical Methods in Biomedical Image Analysis, 1996., Proceedings of the Workshop on*, pages 160–169, 1996.

[20] Chandrasekhar Pisupati, Lawrence Wolff, Wayne Mitzner, and Elias Zerhouni. Geometric tree matching with applications to 3-d lung structures. pages 419–420, 1996.

[21] Jeremy G. Siek, Lie Quan Lee, and Andrew Lumsdaine. *The boost graph library: user guide and reference manual.* 2002.

[22] W. Tang and Albert Chung. Cerebral vascular tree matching of 3-d-ra data based on tree edit distance. *Medical Imaging and Augmented Reality*, pages 116–123, 2006.

[23] J. Tschirren, G. McLennan, K. Palagyi, E. A. Hoffman, and M. Sonka. Matching and anatomical labeling of human airway tree. *IEEE Transactions on Medical Imaging*, 24(12):1540–1547, Dec 2005.

[24] Juerg Tschirren, Eric A. Hoffman, Geoffrey McLennan, and Milan Sonka. Branchpoint labeling and matching in human airway trees. *Medical Imaging 2003: Physiology and Function: Methods, Systems, and Applications*, 5031(1):187–194, May 2 2003.