Computerized Medical Imaging and Graphics 34 (2010) 605–616

Contents lists available at ScienceDirect

Computerized Medical Imaging and Graphics

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

VascuSynth: Simulating vascular trees for generating volumetric image data with ground-truth segmentation and tree analysis

Ghassan Hamarneh*, Preet Jassi

Medical Image Analysis Lab, School of Computing Science, Simon Fraser University, Canada

ARTICLE INFO

Article history:
Received 14 March 2010
Received in revised form 30 May 2010
Accepted 4 June 2010

Keywords:
Simulation
Validation
Medical image analysis
Segmentation
Tubular
Tree-like
Vasculature
Bifurcation
Ground-truth
3D images

ABSTRACT

Automated segmentation and analysis of tree-like structures from 3D medical images are important for many medical applications, such as those dealing with blood vasculature or lung airways. However, there is an absence of large databases of expert segmentations and analyses of such 3D medical images, which impedes the validation and training of proposed image analysis algorithms. In this work, we simulate volumetric images of vascular trees and generate the corresponding ground-truth segmentations, bifurcation locations, branch properties, and tree hierarchy. The tree generation is performed by iteratively growing a vascular structure based on a user-defined (possibly spatially varying) oxygen demand map. We describe the details of the algorithm and provide a variety of example results.

1. Introduction

Medical images provide a wealth of data about internal anatomy essential for computer assisted surgery, computer aided diagnosis, treatment, and tracking of diseases. Images of vasculature and other tree-like structures (e.g. lung airways), in particular, provide important information about the delivery of nutrients to different organs and tissues [1]. Additionally, various pathologies may alter the vasculature. Therefore, the segmentation and analysis of images of vasculature are of utmost importance for understanding, diagnosing, and treating diseases. Examples of analyses performed on the vasculature includes discovering vascular tree topology and branching patterns, calculating radii, lengths, and tortuosity of branches using measures such as the distance metric (DM), the inflection count metric (ICM), and the sum of angles metric (SOAM) [2]. Visual inspection of the image data and manual segmentation and analysis are very time consuming, tedious, suffer from inter- and intra-operator variability and can be subjective and qualitative. This hindrance, in conjunction with the acquisition of large numbers of images of vasculatures, e.g. using phase contrast magnetic resonance angiography or computed tomography angiography, has created a strong need for highly automated, fast, accurate and robust computerized segmentation and analysis algorithms. This strong need has been the catalyst for algorithms for segmentation and analysis of vasculature. This has been reflected by the large number of proposed algorithms for enhancement, segmentation, and analysis of vasculature and perfusion data from 3D medical images in the past few decades [3–12]. However, what has been lacking is the existence of benchmarks for the validation, evaluation, and comparison of all these approaches. Unfortunately, a database of ground-truth segmentation and analyses (branch point locations, tree-hierarchy, etc.) of vasculature does not currently exist, which makes validating and benchmarking such algorithms, as well as training machine learning techniques, very difficult. There are, however, databases or projects for validation of data other than tubular branching trees; most notably are BrainWeb [13], the Internet Brain Segmentation Repository (IBSR),[1] PET-SORTEO [14], the Non-Rigid Image Registration Evaluation Project (NIREP) [15], VALMET [16], and STAPLE [17]. This is where our work is most relevant. It is a unique validation project that generates data in a controlled manner allowing for extensive validation of existing vasculature segmentation and analysis algorithms. In particular, we

* Corresponding author. Tel.: +1 778 782 3007; fax: +1 778 782 3045.
E-mail addresses: hamarneh@cs.sfu.ca (G. Hamarneh), preetj@cs.sfu.ca (P. Jassi).

[1] Internet Brain Segmentation Repository: http://www.cma.mgh.harvard.edu/ibsr/.

0895-6111/$ – see front matter
doi:10.1016/j.compmedimag.2010.06.002

simulate vascular trees embedded into volumetric images and generate the corresponding ground-truth segmentation (labeling), tree topology, and other measures that are essential for benchmarking and validation of the 3D vasculature segmentation and analysis algorithms.

There are two main methods for simulating vascular structures: (i) Lindenmayer systems: a set of production rules (grammar) used to iteratively generate complex shapes, often used to describe the growth of plants [18]; and (ii) iterative growth into a perfusion volume: growing a vascular structure by iteratively connecting new terminal nodes chosen from some volume[19–22]. We follow the latter approach because it allows greater control over the volume of the vasculature and is amenable to conforming to physical hemodynamics laws and blood vessel formation constraints, as we will see later.

In this work, we simulate realistic vascular trees based on user-defined (possibly spatially varying) demand for nutrients. The trees are then used to create volumetric images, which may then be customized to simulate a particular medical image acquisition modality. The result is the generation of volumetric images of vasculature, their corresponding ground-truth segmentations, bifurcation locations in 3D, and formal descriptions, using Graph eXchange Language (GXL), of the vascular tree hierarchy and branch attributes such as radii, flow, and lengths.

In the remainder of the paper, we describe the details of the underlying physics-based vascular model in Section 2. In Section 3, we describe the vasculature generation algorithm that we adopted, which is initialized with a user-defined oxygen (nutrient) demand map and other tunable parameters (Sections 3.2–3.3). We then describe the iterative tree-construction algorithm for selecting terminal nodes, creating bifurcating branches, calculating branch radii and flow, and updating the supply of nutrients (Sections 3.4–3.10). We describe the graph representation in Section 3.11 and the creation of volumetric images in Section 3.12. We follow with comprehensive results and examples in Section 4, and concluding remarks in Section 5.

2. Vasculature flow model

We begin, in this section, by describing the underlying physics-based vascular model that governs the way that the vascular structure is generated. Specifically, we discuss the flow and radii constraints of a branch in the vascular structure. Additionally, we present equations to calculate the reduced resistance as well as the pressure drop along a branch in the vascular structure. The flow and radii constraints, as well as the reduced resistance equations, are used to determine the radii of the branches in the generated vascular structure. The equations and constraints presented here will be used in later sections when developing the iterative vascular generation.

Two flow constraints in a vascular network are of particular importance for the simulation: (i) Conservation of flow, under which the flow entering a bifurcation point through the parent Q_{parent} must equal the total flow leaving a bifurcation point via the 'left' and 'right' child branches, Q_{left} and Q_{right}, i.e.

$$Q_{parent} = Q_{left} + Q_{right}. \tag{1}$$

(ii) The flow into each terminal node Q_{term} must be the same[2] and equal to the perfusion flow Q_{perf} at the root of the tree divided by the total number of terminal nodes N, i.e.

$$Q_{term} = \frac{Q_{perf}}{N}. \tag{2}$$

The radius of the parent branch r_{parent} is related to the radii of its two children, r_{left} and r_{right}, as follows

$$r^{\gamma}_{parent} = r^{\gamma}_{left} + r^{\gamma}_{right} \tag{3}$$

with γ set between 2.55 and 3 [20,23]. The flow resistance R_i induced along a branch i with radius r_i and length L_i is

$$R_i = \frac{8\eta L_i}{\pi r_i^4} \tag{4}$$

where η is the viscosity of the fluid, which is assumed constant in this work for simplicity. The reduced resistance R^* for a terminal branch[3] *term* is obtained by substituting $r = 1$ in (4),

$$R^*_{term} = \frac{8\eta L_t}{\pi} \tag{5}$$

R^* for a bifurcation sub-tree with (non-terminal) branch i as its root is given by [20]

$$R^*_i = \frac{8\eta L_i}{\pi} + \left[\frac{(r_{left}/r_i)^4}{R^*_{left}} + \frac{(r_{right}/r_i)^4}{R^*_{right}} \right]^{-1} \tag{6}$$

As we will see later, the reduced resistance is used for calculating the radii of branches. The pressure drop along a branch i is calculated as

$$\Delta P_i = Q_i R_i \tag{7}$$

Each terminal node is assumed to have the same terminal pressure, which means the pressure drop from the perfusion point to any terminal node must be the same. Since the flow at each terminal node is the same and likewise the terminal pressure, then the resistance from the perfusion point to any terminal node must also be the same.

3. Vascular tree generation

3.1. Overview

The tree generation is performed by iteratively growing a vascular structure based on a user-defined (possibly spatially varying) oxygen demand map (Section 3.2) while enforcing the physical constraints (Section 2). At a high level (Fig. 1), during each iteration, a candidate terminal node is first chosen according to the oxygen demand map. Then an existing branch is bifurcated to supply this terminal node. The choice of the branch to bifurcate and the location of the bifurcation point along the branch is chosen in an optimal way as we explain later. The oxygen supply map is then updated accordingly, and a new terminal node is selected, and so on. After the tree has completed its growth, the final radii of the tree branches are calculated according to the aforementioned physical flow and radii constraints.

3.2. Oxygen demand map

An oxygen demand map $ODM(x, y, z) : \Omega \subset R^3 \rightarrow [0, 1]$ is a 3D scalar volume that covers the entire perfusion volume Ω, with values ranging from 0 for 'no demand' to 1 for maximum demand for oxygen. *ODM* is provided as an input to the algorithm. A simple

[2] This assumption avoids having varying terminal flows and subsequently avoids the knapsack combinatorial optimization problem of choosing candidates.

[3] We assume a cut-off in the resolution of our model and hence ignore the capillary loops bringing blood back to the heart. With this assumption we assign a resistance even to the terminal branches of our model.