UC San Diego

UC San Diego Electronic Theses and Dissertations

Title

Dense Image-to-Image and Volume-to-Volume Labeling

Permalink

https://escholarship.org/uc/item/2zs578mq

Author

Merkow, Jameson Tyler

Publication Date

2017

Peer reviewed|Thesis/dissertation

eScholarship.org Powered by the California Digital Library
University of California

UNIVERSITY OF CALIFORNIA, SAN DIEGO

Dense Image-to-Image and Volume-to-Volume Labeling

A dissertation submitted in partial satisfaction of the
requirements for the degree of Doctor of Philosophy

in

Electrical Engineering (Signal & Image Processing)

by

Jameson Tyler Merkow

Committee in charge:

Professor David Kriegman, Chair
Professor Truong Nguyen, Co-Chair
Professor William Hodgkiss
Professor Gert Lanckriet
Professor Zhuowen Tu

2017

Copyright

Jameson Tyler Merkow, 2017

All rights reserved.

The Dissertation of Jameson Tyler Merkow is approved and is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____

_____
Co-Chair

_____
Chair

University of California, San Diego

2017

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like express my sincere gratitude to my advisers Professor David Kriegmen and Professor Zhuowen Tu for the continuous support of my Ph.D study and research. Professor Kriegmen's patience, motivation and experience pushed me to complete this thesis and my Ph.D study. Professor Tu supported me in every way possible and always pressed me to exceed my own expectations for my work. I would not have completed my study without his knowledge, expertise and guidance. Though Professor Kriegman is now listed as my official adviser, that was not always the case, I want to thank him and Professor Tu for always looking out for what is best for my career.

My sincere thanks go to the rest of my committee, Professor Gert Lanckriet, Professor William Hodgkiss, and Professor Troung Nguyen for their encouragement, insightful comments, and hard questions.

I would also like to thank Professor Alison Marsden for supporting my work for many years and encouraging me to seek every opportunity, both academic and professional. Without her my study would have been over before it started.

Finally, I would like to thank my wife Gisele. Her support, encouragement, quiet patience and unwavering love were undeniably the bedrock upon which the past five years of my life have been built. Her tolerance of my occasional vulgar moods is a testament in itself of her unyielding devotion and love. I thank my parents, Leslie Jameson and Alan Merkow, for encouraging me to follow my dreams and ambitions. It was under their watchful eye that I gained the drive and ability to tackle challenges head on.

Chapter 2, in part, is a reprint of the material as it appears in Jameson Merkow, Zhuowen Tu, David Kriegman, and Alison Marsden. International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2015. The dissertation author was the primary author of this paper.

Chapter 3, in part, is a reprint of the material as it appears in Jameson Merkow, Alison Marsden, David Kriegman, and Zhuowen Tu. International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2016. The dissertation author was the primary author of this paper.

Chapter 3, in part, has been submitted for publication of the material as it may appear in Jameson Merkow, Alison Marsden, Zhuowen Tu, and David Kriegman. Medical Image Analysis, 2017 The dissertation author was the primary author of this paper.

Chapter 4, in part, is currently being prepared for submission for publication of the material as it may appear in Gabriel Maher[1], Jameson Merkow[1], David Kriegman, and Alison Marsden. The dissertation author was one of two equal contributing authors of this paper in both algorithm and manuscript development.

[1]Equal contribution

VITA

2003-2007	Bachelor of Science in Computer Engineering, Tulane University School of Engineering, New Orleans, LA
2007-2009	System Integration and Test Engineer, Raytheon, Marlborough, Ma.
2009-2010	Masters of Science in Engineering, Carnegie Mellon University, Pittsburgh, PA.
2010-2017	Doctor of Philosophy, University of California, San Diego.

PUBLICATIONS

Jameson Merkow, Brendan Jou, and Marios Savvides. "An exploration of gender identification using only the periocular region." Biometrics: Theory Applications and Systems (BTAS), 2010

Hongzhi Lan, Jameson Merkow, Adam Updegrove, Daniele Schiavazzi, Nathan Wilson, Shawn Shadden, and Alison Marsden. "Simvascular 2.0: An integrated open source pipeline for image-based cardiovascular modeling and simulation." American Physical Society, Division of Fluid Dynamics Meeting Abstracts, 2015.

Jameson Merkow, Zhuowen Tu, David Kriegman, and Alison Marsden. "Structural edge detection for cardiovascular modeling." International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2015.

Jameson Merkow, Alison Marsden, David Kriegman, and Zhuowen Tu. "Dense volume-to-volume vascular boundary detection." International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2016.

Adam Updegrove, Jameson Merkow, Daniele Schiavazzi, Nathan Wilson, Alison Marsden, and Shawn Shadden. "Development of an open source image-based flow modeling software-simvascular." American Physical Society Division of Fluid Dynamics Meeting Abstracts, 2014.

Adam Updegrove, Nathan M Wilson, Jameson Merkow, Hongzhi Lan, Alison L Marsden, and Shawn C Shadden. "Simvascular: An open source pipeline for cardiovascular simulation" Annals of Biomedical Engineering, 2016

Jameson Merkow, Alison Marsden, Zhuowen Tu, and David Kriegman. "3D Convolutional Neural Networks for dense volume-to-volume Labeling." *Invited and under review at* Medical Image Analysis, 2017

Gabriel Maher[2], Jameson Merkow[1], David Kriegman, and Alison Marsden. "DeepLofting: building 3D cardiovascular models with convolutional neural networks." *To be submitted to* IEEE Transactions on Biomedical Engineering, 2017.

Jameson Merkow, Robert Luftkin, Kim Nguyen, Stefano Soatto, Zhuowen Tu, and Andrea Vedaldi. "DeepRadiologyNet: Radiologist Level Pathology Detection in CT Head Images." *Being prepared for publication.*

[2]Equal contribution

ABSTRACT OF THE DISSERTATION

Dense Image-to-Image and Volume-to-Volume Labeling

by

Jameson Tyler Merkow

Doctor of Philosophy in Electrical Engineering (Signal & Image Processing)

University of California, San Diego, 2017

Professor David Kriegman, Chair
Professor Truong Nguyen, Co-Chair

This thesis presents three principled approaches to dense pixel and voxel level labeling and demonstrates their effectiveness for both segmentation and boundary detection and subsequent use in 3D model construction from volumetric data.

First, a structured decision tree classifier for vessel wall segmentation on volumetric angiograms is presented. Building upon advances in natural image boundary detection, this 3D classifier generates structured patch-to-patch 3D vessel boundary localization using domain specific volumetric features, an adaptive prior coupled with an importance driven sampling scheme. Through comparison of our methodologies to a number of baselines including widely

used edge detection strategies, non-structured decision trees, and approaches using alternative input features the effectiveness of the classifier is demonstrated. Additionally, the classifier is shown to be robust to error in the *a-priori* information.

Second, a 3D Convolutional Neural Network (CNN) approach to pixel classification is introduced. Two CNN classifiers are presented, HED-3D and I2I-3D. The first extends the popular Holistically-Nested Edge Detector into 3D to perform generic volumetric segmentation. A second 3D CNN is introduced that performs precise localization using a novel fine-to-fine, multi-scale architecture. This classifier addresses three key issues to precise image-to-image and volume-to-volume labeling: 1) efficient end-to-end voxel label prediction and training, 2) precise localization capable of capturing fine structures typical in medical data, and 3) direct multi-scale, multi-level representation learning. I2I-3D is shown to outperform alternative fine-to-fine strategies through demonstration and evaluation on multiple data-sets and tasks. We evaluate these frameworks on three challenging tasks, vessel boundary prediction, brain boundary prediction and skull-stripping.

Lastly, this fine-scale localization method is augmented with spatial context processing to perform automatic 3D cardiovascular model construction from medical image data. This approach builds upon the I2I architecture to generate accurate segmentation as part of DeepLofting, an efficient pipeline for 3D cardiovascular model construction. The I2I classifier is extended to use spatial context during prediction, forming a new classifier I2I-FC. This powerful classifier is a critical component in DeepLofting, which builds 3D cardiovascular models from medical volumes and *a-priori* information. DeepLofting is evaluated on a publicly available cardiovascular model dataset, and represents a critical step forward in 3D model generation.

Chapter 1

Introduction

1.1 The Role of Pixel Level Labeling

Understanding and interpreting visual data is an essential precursor to almost any intelligent system. A core component to this understanding is the ability to differentiate and classify input images, video and other visual data. Pixel level labeling is the process where pixels or neighborhoods of pixels are classified into a finite number of categories. Pixel level labels can take many forms, including segmentation, where each pixel is given a category to differentiate its neighbors, contour detection, where the boundary between objects are labels, or detection where regions of an image are identified to contain a particular object.

1.1.1 Deterministic Methods

Early approaches to pixel level labeling used deterministic methods, where both algorithmic features and the classifier were manually designed. Many of these classifiers have remained popular including various edge operators such as the sobel, prewitt and laplacian filters as well as the Canny edge detector. Filter based methods use specialized convolutional filters to transform pixel intensity values such that perceived boundaries give a high response. The Canny edge detector [Can86] remains one of the most widely used edge detectors for 2D and 3D vision. Canny uses local gradient information in a multi-step process. First, the image is smoothed using a Gaussian blur with a specified width (variance), and image gradients are

calculated. Second, non-maximum suppression (NMS) removes superfluous responses. Last, hysteresis thresholding is applied such that connected edges are retained, and orphan edge responses are discarded.

Many deterministic segmentation approaches such as Ostu's method [Ots79] and the watershed transform continue to enjoy wide spread use as well. Otsu's method remains among the most popular threshold based segmentation techniques. Otsu threshold clusters pixels into classes by calculating thresholds which optimally separate pixel values such that their intra-class variance is minimized. Otsu can be thought of as a one-dimensional fisher's linear discriminant analysis (LDA) [Fis36]. Another popular morphological segmentation method, the watershed transform, treats images as a topographical map, where intensity values represent 'elevation'. The watershed transform identifies 'catchment basin' and 'watershed ridge lines' which simulate flooding the geographical topology. In this way, basins become contiguous regions and watershed ridges represent the boundaries between regions.

1.1.2 Statistical Models

Over the years, segmentation and boundary detection evolved to use more advanced statistical models to categorize pixel labels. One such method, region growing, selects pixels by comparing values among pixel neighborhoods to determine membership. Starting from an initial selection, pixels along the border of the pixel neighborhood are compared using a statistical analysis and other similarity criteria to the currently selected group, those that are sufficiently similar are added. This process is repeated to iteratively 'grow' the region [OC89, TB97, SC05]. Graph cut algorithms use statistical similarity on a global level to partition pixels into distinct parts. One popular partition criterion, normalized cuts, uses a normalization factor based on node connectivity to optimally partition graphs [Shi00, MBLS01].

Deformable models use combined image statistics (external forces) and geometric

constraints (internal forces) to build consistent models. Internal forces are use geometric properties of the model such as curvature. External forces are composed of image based forces such as edge attraction (advection) or data clustering. These two forces act in opposition of each other to grow from a seed point while adhering to these constraints. One of the most popular deformable model approaches are active contours or 'snakes'. Active contours come in many names including, snakes, active surfaces, balloons, and deformable contours. In general, there are two type of active contours, parametric and geometric which differ in their representation of their curves and surfaces. Parametric snakes represent their surfaces and curves using an explicit, parametric forms. This representation allows direct interaction and manipulation of the model. While this restrictive model topology makes them easier to implement, splitting and merging curves becomes difficult. In contrast, geometric models use an implicit representation, allowing model topology adaption to occur naturally. Level sets [OS88] are among the most popular type of geometric deformable model and have been applied to a great deal of problems [Set96, Wan01, MS89, VC02, LHD$^+$11, NC14, VC02].

1.1.3 Learning-Based Methods

As data-sets improve and large annotated data-stores became prevalent, so did learning based methods. Learning based methods use a set of features, usually derived from pixel intensity values, and image annotation to train a classifier to detect or classify pixel labels. Many algorithms use complex features and a simple classifier for pixel level label prediction. For example, [MFM04] used a large set of pre-defined features, including oriented energy and gradient-based edge features, all from a range of orientations and scales as input to a simple weighted aggregation classifier for object boundary prediction. [DTB06] built upon this idea by using a much more powerful learning component, probabilistic boosting trees (PBT) [Tu05]. [DTB06] proposed a learning based classifier that uses many feature channels to train a PBT for detection of edge aligned pixels. Decision tree based approaches, such as [DTB06], have

been shown to be particularly effective when combined with a structured output [DZ13, ZD14, DZ15, MTKM15]. These "structured forest" classifiers build upon PBTs but classify entire regions/patches rather than single pixels. These patch-to-patch methods have been shown to be effective in a number of applications [MTKM15, MPTAVG16, SLC$^+$14, BBH$^+$16].

Alternative approaches, such as [AMFM11, RB12] use complex features and a simple classifier to extract object boundaries and segmentation. The gPb classifiers [AMFM11] use multiple local and global cues with graph cuts and a simple weighting approach to simultaneously localize boundaries and object segmentation. Another strategy, used sparse code gradients [RB12] to build complex features using K-SVD [AEB06, RZE08] for an SVM-based classifier contour detector.

The continued success of learning based approached led to many new and powerful feature extraction techniques. Feature extraction is used to capture a multitude of contextual cues including gradient direction [DT05, KM08], spatial in-variance [Low04], pixel difference [VJ04, OPM02]. These features are also commonly combined into sets such as textons [LM01, MBLS01] or integral feature channels [DTPB09].

1.1.4 Deep Learning and Convolutional Neural Networks

The success of learning based methods led to those that learn both features and classifiers. This class of methods is often referred to as "deep learning". Deep learning is currently one of the most widely used methods, across a multitude of fields including computer vision. Deep learning systems simultaneously learn complex image features and decision boundaries to produce robust classifiers that can be applied to multiple computer vision tasks.

What is now known as deep learning, has its roots in several innovations across nearly 60 years. Early neural networks using "thresholded logic units", first theorized in 1943, were designed to mimic the way a neuron works [MP43] which eventually evolved into

perceptron units [Ros58] in the late 1950s. Years later, in 1989, modern back-propagation [LBD+89, LBD+90] was theorized by showing that neural nets with many hidden layers could be trained with a simple procedure for digit classification on the popular MNIST dataset [LCB10]. Nearly 20 years later, Geoff Hinton introduced unsupervised pre-training along with deep belief networks (DBNs) to learn, both, high and low level features on an unsupervised task [BLPL07]. The weights learned for the unsupervised task were then re-purposed for training an expanded network on a supervised task with excellent accuracy.

The renewed interest in the re-branded "Deep Learning" classifiers led to a break through in 2012 on the Large Scale Visual Recognition Challenge (LSVRC) [DDS+09]. To this point, top models had errors between 28% and 26% on LSVRC, however with a deep learning based submission by Alex Krizhevsky, Ilya Sutskever, and Geoff Hinton, AlexNet, the error in this challenge was cut nearly in half to 16% [KSH12]. AlexNet combined many breakthroughs which would become main-stays of deep learning networks, including the use of GPUs for fast computation, and rectified linear units (ReLUs).

Deep Learning has become a powerful tool for not only in image classification but also natural language processing [Kim14], speech recognition [HCC+14], bioinformatics [LXLF14], and many other fields [MKS+13, LLS15, DY14]. With the development a variety of deep learning and convolutional neural network (CNN) libraries [DJV+13, ARAA+16, JSD+14, AAB+15, CLL+15, CKF11] and fierce competition on many high quality data-sets such as [MFTM01, DDS+09, EVGW+10, LMB+14], the field has had great number of breakthroughs in optimization techniques [Zei12, KB15, DHS11], regularization techniques [SHK+14, GWFM+13, LXG+15, IS15] and improved architectures [SZ14, SLJ+15, HZRS15].

After the success of deep learning and CNNs in image classification, it was not long until these powerful tools were used for pixel level labeling. With noteable advances in scene labeling [FCNL13, HAGM14b], object segmentation [GGAM14, GL14], detection [SVL14,

Gir15] and boundary detection [SWW+15, BST15]. Fully convolutional neural networks [LSD15] provided simultaneous performance and accuracy in semantic segmentation which led to multiple adaptations that are top performers in a number of pixel labeling applications [XT15, RFB15, MMKT16, CPK+14].

1.2 Medical Imaging

Though the methods and concepts in this work can be generalized to natural images, many of the implementations are applied to medical imaging. Here, we outline the basics of medical imaging and provide an overview of cardiovascular model building, an application that appears throughout this thesis.

Medical imaging is the process and technique of generating visual representations of the interior of the body, typically for a clinical analysis or another medical purpose. These methodologies, referred to as imaging modalities, use a multitude of imaging technologies and are categorized in a number of ways. Many medical imaging techniques fall into one of two categories: projection or tomographic.

Typical projection images, such as radiographic, emit x-ray beams through an object (patient) onto a a detector. Based on the absorption, scatter and other physical characteristics, an image is formed by the detector. These images are two dimensional presentations of 3D objects where pixel intensities depend on the object between the emitter and the detector.

Tomography images are formed by imaging sections/slices which are joined together to form an image volume. Slices may be collected through many different methods, for example, computed tomography (CT) slices formed using x-rays, or magnetic resonance (MR) which uses magnetic fields to generate images.

Tomographic medical image volumes use three anatomical planes (axial, sagittal, and coronal) to describe location or movement of anatomy and anatomical structures. The axial (or traverse) plane perpendicular to the line drawn from the head to the toe. An axial plane

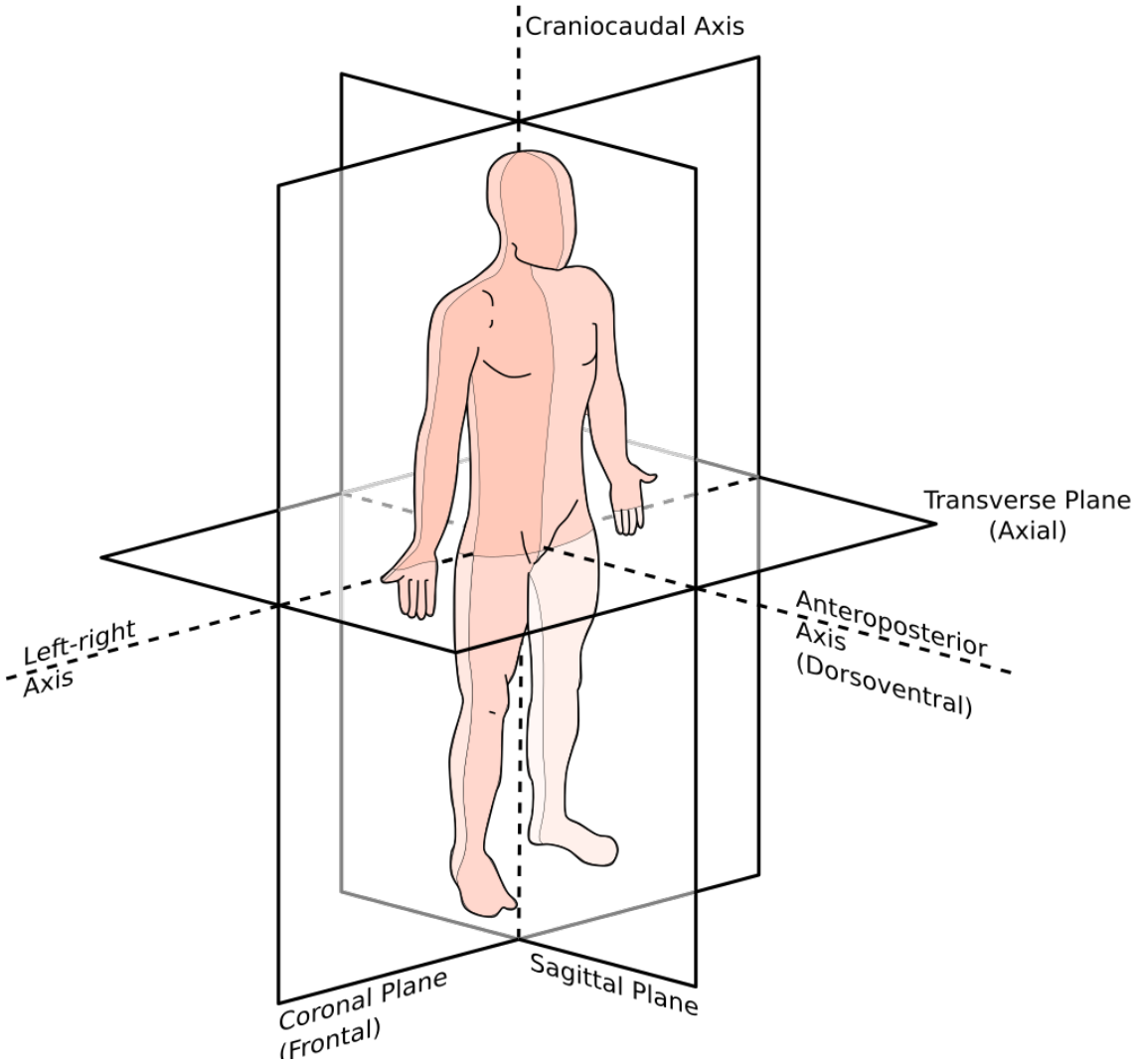


Figure 1.1. Illustration of the three anatomical places: axial (traverse), sagittal, and coronal.

that is closer to the head is called *superior*, where as *inferior* axial planes are those closer to the toe. The sagittal plane divides the volume into *right* and *left* sections from ear to ear. Lastly, the coronal plane is orthogonal to the line passing from the navel through the back, separating *anterior* planes from the *posterior* planes. Figures 1.1 and 1.2 depict illustrations of these three planes.

Since medical image volumes represent consistent physical and anatomical structures they are often coupled with meta-data describing the conversion from pixel space into physical

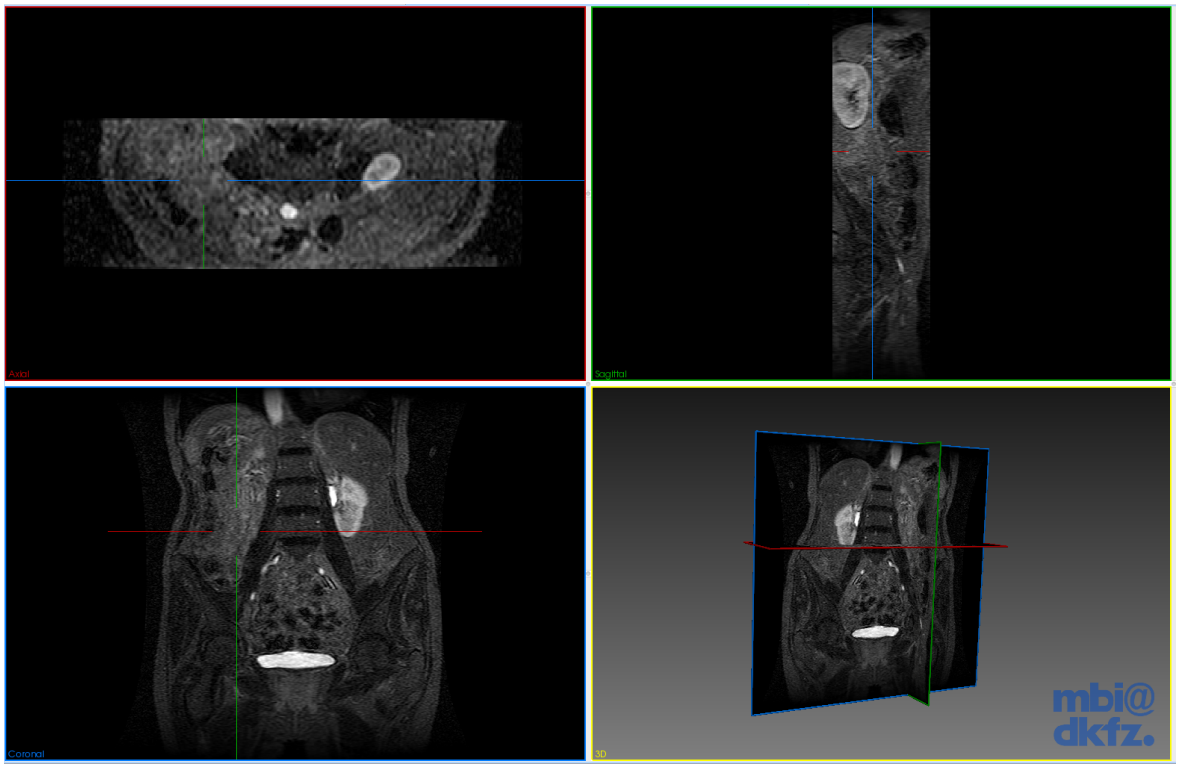


Figure 1.2. Depiction a medical volume showing images in the three anatomical planes. From top-left to bottom-right: Image taken in the axial place, the saggital plane, the coronal plane and a 3D rendering of the intersection of the three planes.

space. Pixel spacing denotes the physical distance between the center of pixels or voxels along all three dimensions. For example, a particular volume may have pixel spacing of $(0.25\text{mm}, 0.3\text{mm}, 1\text{mm})$ meaning there is 0.25mm between pixel columns, 0.3mm between rows, and 1mm between aisles. Spacing is important for rendering medical volumes to ensure that information is displayed proportionally, and for analysis as the physical distance between voxels have bearing on the underlying structures. It is also common normalize volumes by re-sampling with predefined spacing values via interpolation.

Though medical imaging is free from much of the nuisance variability common in natural imaging, such as illumination, scale or occlusion, it does come with its own set of unique challenges. For example, typical medical imaging volumes contain partial voluming, where low acquisition resolution causes multiple tissues or structures to contribute

to individual pixel or voxel intensity values. This makes precise segmentation of medical images more difficult. Another major challenge in medical imaging is the size of the data itself. A typical volume is built from hundreds or thousands individual slices.

1.2.1 Pixel Level Labeling in Medical Images

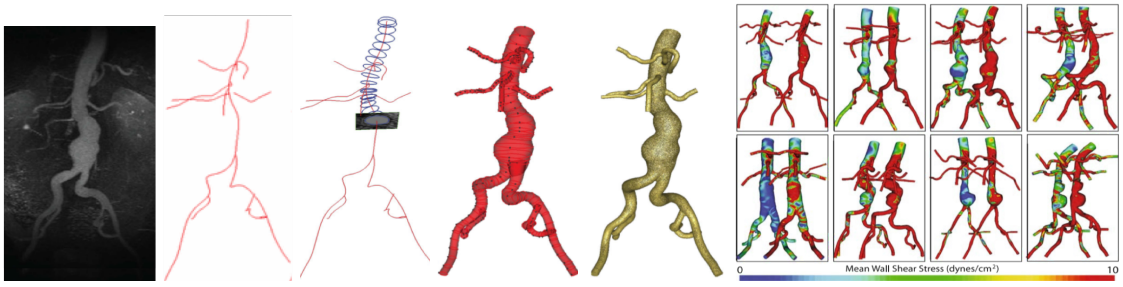
Many of the methods mentioned in Section 1.1 have been directly applied or modified for use in medical imaging.

Thresholding is a simple yet effect method for medical image segmentation that has remained popular for many years. Usually performed interactively, thresholding is sometimes used as a precursor to a more advanced method [SP97] or used on its own [LHKU98, LKC+95]. A major limitation of thresholding, is that it does not take into account physical characteristics, a property that is especially important in medical imaging.

Region growing is another a mainstay in medical imaging segmentation. Medical images and volumes have sparse and connected regions of interest, making an algorithm that sparsely labels pixels (starting from a initial point of interest) both effective and efficient. Region growing has been used for a variety of tasks, including tumor detection in MR images [GBBH96], classification of mammograms [PPO+96] and multi-modal usages [PT01]. The watershed transform is another popular method adapted to medical image segmentation. 3D versions of this algorithm have been applied to various modalities including CT data [WHOF96, SAB05] and MR data [SSV+97, BMHA00, GMA+04]. It is popular approach to use in 2D medical segmentation as well [XLS11, KSG09].

Medical imaging has used a variety of approaches based on graph cuts and spectral clustering as well. These methods utilizes methods ranging from basic clustering techniques such as fuzzy c-mean [PP99] to self-organizing maps (SOM) for clustering pixels in images and for segmenting volumes [MX98, AF97].

Deformable models are commonly used in medical imaging [MT96, LABFL09].



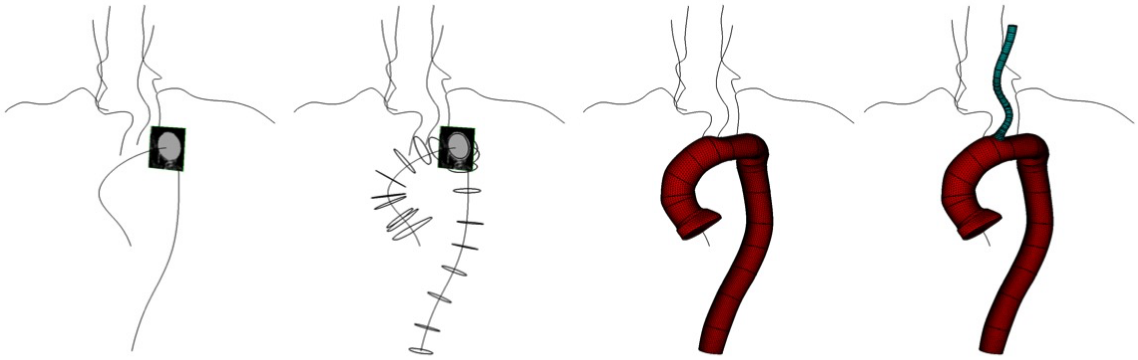
(a) **(b)** **(c)** **(d)** **(e)** **(f)**

Figure 1.3. The cardiovascular model construction work-flow used in SimVascular [UWM+13]. Starting from (a) Image data, (b) users manually generate pathlines, (c) use these pathlines to segment 2D cross section contours, (d) loft segmented contours into a 3D model, (e) generate a numerically stable 3D geometric mesh, and finally (f) computation flow simulations are calculated. Figure reproduced from [UWM+13].

These types of models are more computationally efficient than classification of entire image/volumes and perform consistently on images with partial voluming artifacts. Deformable models are among the most popular methods used for segmentation in, both, 3D volumes [JM97, CDPY99, YPC+05, YSD04] and 2D medical images [NC14, LXGF10, Wan01].

Learning based classifiers are also prevalent in medical imaging such as decision trees [LBW+08, ZLG+11, WLW+15, SHW+15] and support vector machines (SVM) [LSJ+06, CPJ06, ENYW+02]. More advanced classifiers such as Tu's auto-context model [TB10] have also been used with great success.

Given that orientation and scale variability is more tightly controlled in medical imaging, atlas information is often used to aid in classification. One such example [FSB+02], marked an early success in full brain segmentation of MRI images by complementing a intra-class statistical models with atlas information, Another approach [RRM04], used a competing classifier framework that applies an atlas model to simultaneously label anatomical regions in CT images.



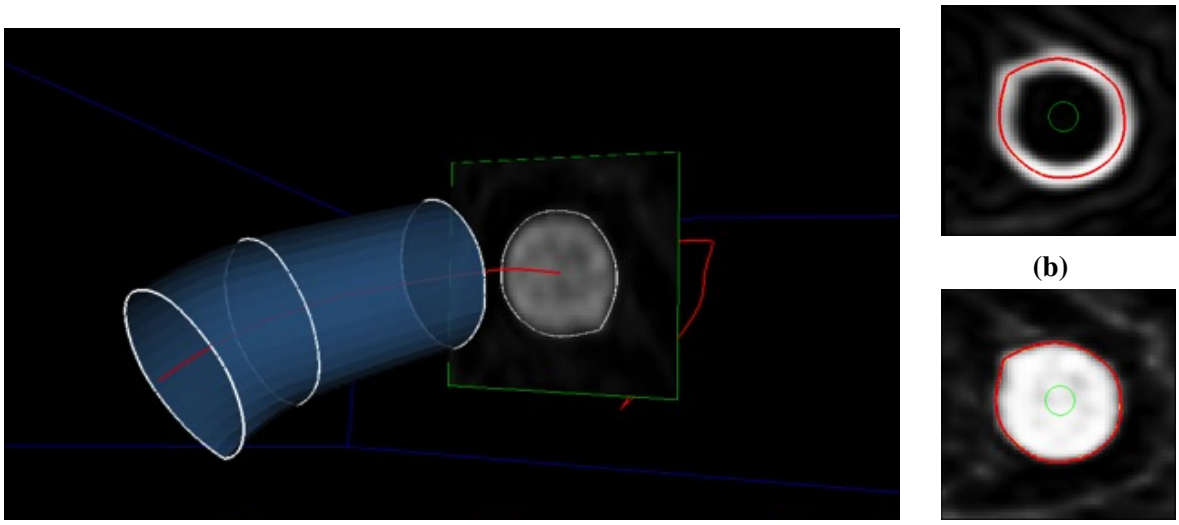
(a) Path creation. **(b)** Cross section segmentation. **(c)** Lofting segmentations. **(d)** Union of lofted surfaces.

Figure 1.4. Illustration of lofting process. Creation of a vascular geometry using the lofted 2D segmentation approach involves (a) creating path to navigate and create a series of segmentations (b) that are lofted to form each vessel (c). A solid model is generated by the union of individual vessel models (d).

1.3 Cardiovascular Modeling

Cardiovascular disease is among the leading causes of death in the modern world [Mar14]. Patient-specific simulations of cardiovascular hemodynamics [TF09] are a groundbreaking tool to improve treatment and diagnosis of various cardiovascular diseases [Mar13]. Hemodynamic simulations use advanced computational fluid dynamics to model blood flow of individual patients. These numerical methods have enabled realistic representations of cardiovascular physiologies. These simulations can improve imaging methods, predict surgical outcomes, localize at risk anatomies and provide increasingly detailed medical data, all at little to no risk to the patient. Hemodynamics simulations have been used to analyze formation of atherosclerotic plaques [SEM+11], develop novel surgical approaches for treatment of congenital heart disease [EMHMoCHAMI15], and as an accurate diagnostic tool for coronary heart disease [TFM13].

Image based hemodynamic simulations were first developed in the late 1990s and early 2000s and have proven to be an important tool for clinical research [NOV11, MAI99,



(a) **(b)** **(c)**

Figure 1.5. An illustration of a typical lofting procedure. As shown in (a), Users navigate along the path, segmenting vessel cross sections, shown in (b) and (c). These segmentations are oriented in 3D space and a 3D surfaces ins generated (shown in (a).

WJV$^+$03, DCFF13, KBBP$^+$14, MKK$^+$16]. HeartFlow was among the first to produce FDA-approved products based on these concepts and they helped to simultaneously reduce the cost and risk to patients with various coronary artery diseases. Most image-based modeling applications start with 3D MR or CT angiograhic data. Image processing is used to generate a 3D geometric model of the vascular regions. This 3D model is imported into a computational flow dynamics (CFD) package where a volumetric mesh is extracted and hemodynamic simulations are carried out to numerically simulate blood flow.

The first step in image-based modeling is to segment the image data in a region of interest (ROI) to extract the boundary or structure of an object from the intensity field. While many image processing and segmentation processes exist, most are not well-suited to CFD simulation which require smooth, noiseless boundaries for accurate simulation. The segmentation process is most commonly used to identify the luminal surface of a blood vessel; however other anatomical structures may be similarly segmented and modeled.

Cardiovascular models can be constructed manually, using medical imaging software,

or by using a variety of image segmentation algorithms to produce a segmentation that can later be refined into a cardiovascular model of sufficient quality. Most available image segmentation algorithms have method-specific parameters that must be tuned to produce accurate segmentations; a process that is cumbersome for non-expert users and introduces user-variation into simulation results [NUZ00]. Manual model construction is similarly time-consuming and requires expert knowledge of cardiovascular physiology. As a result, cardiovascular model construction is currently a major bottleneck in performing large-scale hemodynamics simulations. Studies on large patient cohorts are required in order to correlate simulation outputs with clinical outcomes.

1.3.1 Cardiovascular Segmentation

In terms of cardiovascular model building methods, approaches generally fall into two categories, 3D and pathline based. Level sets are popular among the 3D approaches, such as [WLK+11] which propagates 3D surfaces through regions of high contrast, and [APB+08] which apply colliding fronts from two seed points set on opposite ends of a blood vessel. Many approaches use image enhancement to make vasculature more visible. These methods range from simple multi-scale pixel intensity based approaches [FNVV98, SNA+97], to complex curvilinear structure enhancement [Law08] and are often combined with level sets to improve performance [LC10, SDN+11]. Connected-ness is an important component to vascular segmentation, making methods that utilize this concept such as graph-based classifiers popular in the field. These methods go beyond pixel intensity, and leverage the tabular structure of blood vessels with shape priors [Bau09], vessel-ness responses [Wan16], Hessian computations [WKN+16], curvilinear structure identification [Tur13, Rob16], and max-flow optimization [Pez16].

Pathline based model construction methods leverage the fact that vascular networks are interconnected tubular, pipe-like structures and navigate along individual vessels to create 2D-

cross section segmentations which can be merged into 3D tubular structures [PTW98, TF09].

This is accomplished by, first, annotating vessel path lines which indicate the path of a single vessel. These pathlines allow the 3D image data to be navigated at oblique angles, where the vessels are relatively centered. Once cross-section segmentations are generated, these contours are oriented in 3D space and a tubular structure is interpolated in a process called 'lofting'. After a number of these lofted structures are created, they are merged through Boolean operations into a single complete 3D model.

Since pathlines are relatively centered in and approximately orthogonal to vessel walls, images extracted along these path lines typically have predictable placement and shape; an example is depicted in Figure 1.5. This extraction technique allows easier parameter selection for geometric constraints and initial contour placement, making active contours and level sets particularly effective [Wan01, LXGF10]. Alternative strategies directly fit curves [Kri00, LY07, MST10, BC11] or templates [ZBG+07, KGPS13] to image data. Other approaches treat the vessel wall as a tracking problem and builds a vessel by updating a segmentation along the pathline from an initial user generated annotation [KYD+17].

1.4 Aims and outline of this work

The remainder of thesis is organized into three main chapters followed by a conclusion. A brief description of each chapter appears below.

1.4.1 Chapter 2 - Structured Forests for Cardiovascular Boundary Detection

Computational simulations provide detailed hemodynamics and physiological data that can assist in clinical decision-making. However, accurate cardiovascular simulations require complete 3D models constructed from image data. Though edge localization is a key aspect in pinpointing vessel walls in many segmentation tools, the edge detection algorithms

widely utilized by the medical imaging community have remained static. In this chapter, we describe an approach to medical image edge detection by adopting the powerful structured forest detector and extending its application to the medical imaging domain. First, we specify an effective set of medical imaging driven features. Second, we directly incorporate an adaptive prior to create a robust three-dimensional edge classifier. Last, we boost our accuracy through an intelligent sampling scheme that only samples areas of importance to edge fidelity. Through experimentation, it is demonstrated that this method outperforms widely used edge detectors and probabilistic boosting tree edge classifiers while being robust to error in *a-priori* information.

1.4.2 Chapter 3 - Convolutional Neural Networks for Dense Volume to Volume Labeling

In Chapter 3, we introduce our approach to dense 3D volume labeling in medical imaging using convolutional neural networks (CNN). To start, we describe an extension of the start-of-the-art classifier, Holistically-Nested Edge Detector (HED) as a new generic volumetric classifier, HED-3D. In addition, we introduce a novel 3D CNN architecture, I2I-3D, that densely labels volumetric data. This fine-to-fine, deeply supervised framework addresses three critical issues to volume-to-volume labeling: (1) efficient, holistic, end-to-end volumetric label training and prediction (2) precise voxel-level prediction to capture fine scale structures prevalent in medical data and (3) directed multi-scale, multi-level feature learning. To show that I2I-3D and HED-3D significantly advance the state-of-the-art, both frameworks are evaluated on multiple labeling tasks across two publicly available datasets where they outperform 2D CNNs, other 3D CNNs, and the current state-of-the-art. We show that these frameworks have state-of-the-art performance on the publicly available dataset LPBA40 by precisely predicting skull stripping masks and brain boundaries with great accuracy. We also evaluate I2I-3D and HED-3D on blood vessel boundary detection with another publicly

available dataset consisting of 93 medical image volumes captured from a wide variety of anatomical regions. In addition, we compare a 2D version of this classifier to HED (2D) and show that our classifier achieves similar performance with greater precision by evaluating on the popular BSDS dataset [MFTM01] with and without non-maximum suppression.

1.4.3 Chapter 4 - Cardiovascular Model Construction with Convolutional Neural Networks

In this chapter, we describe a novel approach in tackling the challenging task of constructing 3D cardiovascular models from medical image data. We combine deep learning with a vessel path-line model construction pipeline to form DeepLofting, a new and efficient method for building cardiovascular models. We start by developing a novel neural network architectures I2I-FC, which augments the I2I architecture with spatial context processing and allows fully convolutional networks to utilize image spatial context to produce accurate localized segmentations. DeepLofting uses convolutional neural networks to compute vessel boundaries along anatomical path-lines and combines these boundaries to form a final cardiovascular model. Given vessel path-lines, DeepLofting is fully-automatic requiring substantially less user-intervention compared to traditional model building workflows. We evaluate our architectures and DeepLofting pipeline on a publicly available dataset of 100 computed tomography (CT) and magnetic resonance (MR) volumes. DeepLofting, combined with I2I-FC, significantly outperforms other neural network architectures and popular cardiovascular model building methods for producing accurate 3D cardiovascular models.

Chapter 2

Structured Forests for Cardiovascular Boundary Detection

2.1 Introduction

Building on advances in medical imaging technology, cardiovascular blood flow simulation has emerged as a non-invasive and low risk method to provide detailed hemodynamic data and predictive capabilities that imaging alone cannot [Mar14]. Blood flow simulations have been used to develop novel surgical methods to treat congenital heart disease, characterize hemodynamics in aneurysms, and assess risk of bypass graft failure [dZMFY10]. A necessary precursor to these simulations is accurate construction of patient-specific 3D models via segmentation of medical image data.

Currently available tools to aid model construction include ITK-SNAP [HCG03] for general medical image segmentation and SimVascular [UWM+13] and VMTK [APB+08] which use specialized segmentation techniques for blood flow simulation. These packages implement automated segmentation tools, most of which rely heavily on region growers, active-contours and snakes. These methods can be effective for cardiovascular segmentation, however they often require finely-tuned algorithm parameters, hence manual segmentation is commonly used in practice. Model creation remains one of the main bottle-necks in widespread simulation of patient specific cardiovascular systems. Increased efficiency of the

model construction process will enable simulations in larger cohorts of patients, increasing the clinical impact of simulation tools. Machine learning techniques produce robust results without manual interaction, which make them a powerful tool for model construction. There has been extensive research into learned edge detection in natural images but edge detection has not received the same attention in medical imaging.

Here, a machine learning based edge detection technique is proposed by leveraging diverse domain specific expert annotations in structured prediction labels and builds upon the structured forest classifier [DZ13]. We contribute a novel combination of domain specific features, introduce *a priori* information, and devise an intelligent sampling scheme to correctly classify edges while using a minimum number of training samples. Much of this chapter was originally published in [MTKM15].

2.2 Background and Relevant Work

Active contours are one of the most widely used strategies for segmentation in medical imaging. Active contours propagate segmentation labels using opposing internal and external forces. Internal forces update the contour via appearance criterion while external forces modulate the contour shape with geometric constraints. This technique enforces contour smoothness making it attractive for vascular segmentation for CFD. However, active contours run into difficulty under varying image conditions and often require finely tuned parameters to prevent segmentations from leaking into background pixels.

Alternative strategies use learning based methods and avoid parameter tuning by leveraging expert annotations to build statistical appearance and shape models. Zheng et. al employed probabilistic boosting trees (PBTs) with pixel transformations, gradient data, and atlas features to quickly generate vessel-ness labels in 46 cardiovascular CT volumes [ZLG$^+$11]. Contextual cues were integrated with PBTs for brain segmentation by using multiple resolution features to learn shape geometry and appearance models in [TND$^+$08]. In

that work, the authors used a discriminative classifier to capture local appearance information, and a generative model to enforce global shape constraints. This hybrid method used very few parameters, and evaluated well qualitatively and quantitatively. Schneider et al. used steerable features with Hough decision forest voting to perform joint vessel and center-line segmentation [SHW+15].

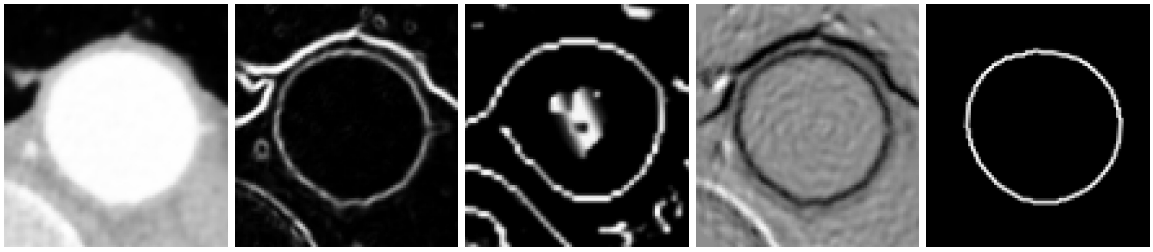


Figure 2.1. Cross sectional appearance feature examples, from left to right: lumen intensity, gradient magnitude, Canny edge map, distance-gradient projection and ground truth.

Research shows that contextual information can boost learned classifier performance. For example, auto context models learn important shape contexts and discriminative features with an iterative approach to segmentation. auto context has been successful in a range of medical imaging tasks including brain segmentation, and segmentation of prostates [TB10, LLF+12]. However, auto context is best suited to regional label maps, making it sub-optimal for contour detection where ground truth is often a single voxel in thickness.

Despite an abundance of research into learned edge detection in natural images, edge detection in medical imaging has remained static. Ren and Bo calculated learned sparse coding features and SVM to localize image contours, successfully combined powerful oriented gradient features and dictionary learning to capture edge characteristics [XB12]. In [LZD13], the authors use discrete sets of clustered local contour features, dubbed *sketch tokens*, to perform various segmentation tasks. Dollár and Zitnick used a clustering algorithm to train a structured forest edge detector [DZ13].

Inspired by the success of contour detection in natural images, we propose a novel edge detection method specific to cardiovascular imaging. We build upon recent work on

structured forests by extending the classifier to work for 3D image patches, defining a novel set of domain specific features, incorporating *a priori* information into classification, and employing a specialized sampling scheme.

2.3 Methods

In this section, we describe the methodology used to train our classifier for edge detection. Our goal is to generate a forest of decision trees that produce high scores at edge aligned voxels, and low scores elsewhere. First, we begin by collecting randomized samples of lumen patches and edge ground truth from a training set of image volumes. Second, we compute feature channels from the input patches and find a discretization of the output space. Last, input features and edge ground truths are used to train a decision tree. This process is repeated N times to generate a forest of N trees. We outline our tree training process in Algorithm 1.

Algorithm 1. Tree training procedure

Given a training set $S = (\mathbf{X}_i, \mathbf{P}_i, \mathbf{Y}_i)$ where $\mathbf{X}_i$, $\mathbf{P}_i$ and $\mathbf{Y_i}$ are the i^{th} training image volumes, atlas priors and edge ground truth.

1: **for all** $(\mathbf{X}_i, \mathbf{P}_i, \mathbf{Y}_i) \in S$ **do**
2: $\quad \mathbf{G}_i \leftarrow \text{ComputeFeatures}(\mathbf{X}_i, \mathbf{P}_i)$
3: $\quad G_c, Y_c \leftarrow \text{CollectSamples}(\mathbf{G}_i, \mathbf{Y}_i)$
4: **end for**
5: $C \leftarrow \Pi_\Phi(Y_c)$ ▷ C is the discrete set of labels $C = \{1, 2, 3, ..k\}$
6: $t \leftarrow TrainTree(G_c, C)$

2.3.1 Sampling Methodology

At each sample location a $16 \times 16 \times 16$ voxel patch and an $8 \times 8 \times 8$ edge ground truth patch are collected. We denote image patch samples as $x \in X$ and boundary annotations as $y \in Y$. Due to computational constraints of collecting 3D samples, fewer training samples could be collected so a specialized sampling scheme was devised. First, all positive samples

were taken from vessel wall aligned overlapping patches to produce more precise edge responses. Though additional structured information can be gathered by including positive sample patches that contain any edge segment, we found that collecting only vessel-wall-aligned samples produced more precise edge responses. Second, negative samples were collected from two zones. We collected a majority of negative samples in the region just outside the vessel wall. However, to properly interpret *a priori* information, approximately 30% of the negative samples were taken from any negative region. These zones were determined based on the vessel wall location supplied by ground truth annotation. Figure 2.2 depicts 2D cross section diagram of our sampling method, we denote the positive sampling zone in blue, the vessel wall adjacent negative sample area in yellow (zone 1) and the general negative sampling region (zone 2) in red.

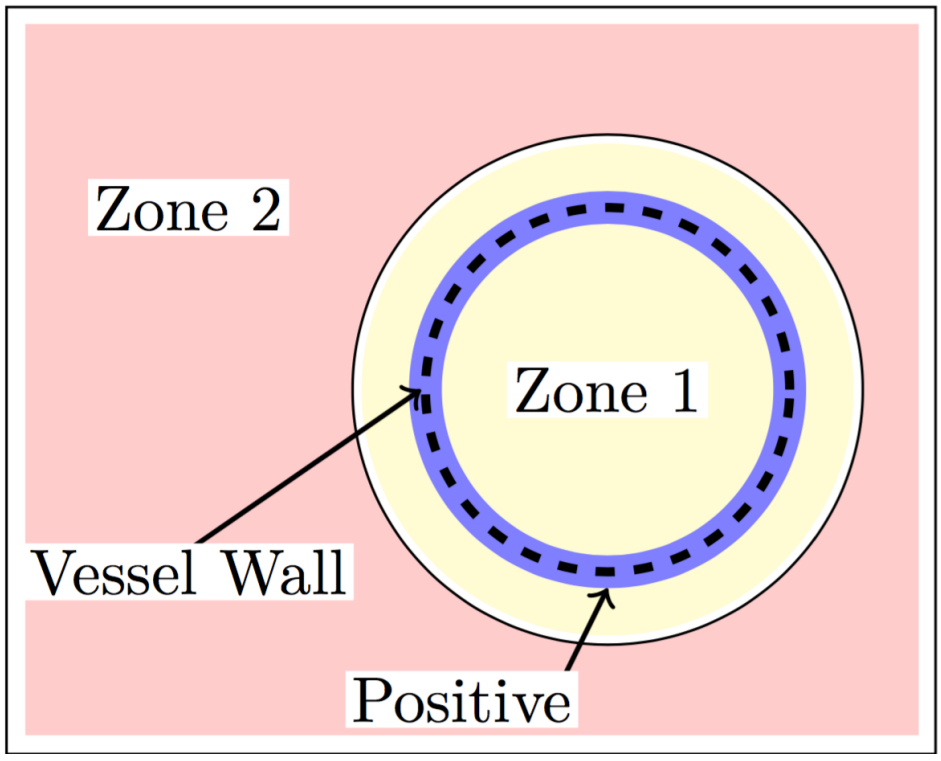


Figure 2.2. Sample zones depicted as a 2D cross section.

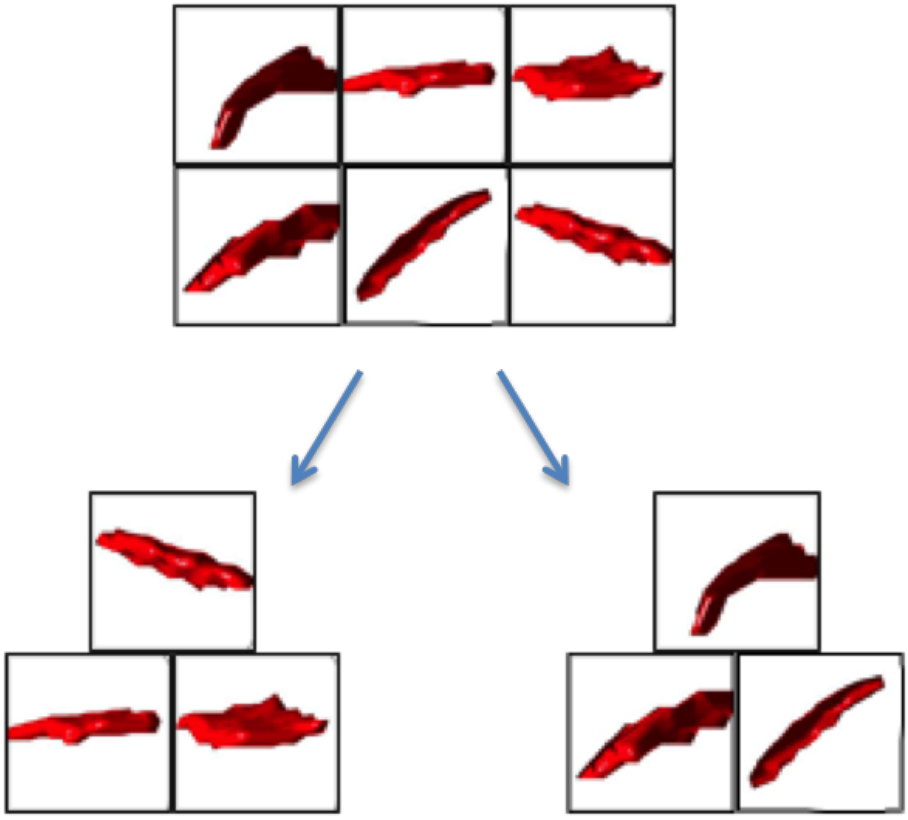


Figure 2.3. Example of structured labels grouped into discrete sets.

2.3.2 Topographical and Image Features

For each sample, a set of appearance and atlas-based feature channels are computed. We selected three appearance features: lumen intensity, gradient magnitude and Canny edge response, each of which are available at full and $\frac{1}{2}$ resolutions. In addition to appearance features, we incorporated *a prior* information by including a distance map from user-generated pathlines as an additional feature channel. Center-lines were generated during model construction by placing points in vessel lumen and connecting them with splines to form interpolated curves. Typically, 15-30 control points are selected and approximately 300 points are generated through interpolation. This feature injects atlas information into the classifier its treated as any other input feature. Next, we combined appearance and *a priori* features into a composite feature by projecting the image gradient onto the gradient of the distance map.

This provides a strong response when the image gradient aligns with the normal direction of the path. Last, difference features were calculated by heavily blurring each feature channel then computing the piece-wise paired differences of five voxel sub-patches.

2.3.3 Edge Detection

Our approach employs a collection of de-correlated decision trees that classify input patches $\mathbf{x} \in X$ with a structured label map $\mathbf{y} \in Y$. This task is presented as a multi-class classification problem where each class is a unique label patch. Decision trees are trained by calculating information gain of the ith element across all j input vectors, $\mathbf{x}_j \in X, \mathbf{x}_j = (x_1, x_2, x_3, ..., x_i)_j$, for labeled samples $y_j \in \mathbf{Y}$. For binary classification, y_j has two possible values, $y_j = \{0, 1\}$, in multi-class systems y_j has k possible values, $y_j = \{1, 2, 3, ..., k\}$. When training decision trees with label *patches*, every permutation of the label patch could be considered a distinct class; this leads to an enormous number of potential classes of y_j. For example, a binary 8×8 label map, would result in $\binom{8\times 8}{2} = 2016$ distinct classes, and this complexity increases exponentially with 3D patches, a $8 \times 8 \times 8$ label patch results in $\binom{8\times 8\times 8}{2} = 130,816$ classes. With a label space of this complexity, information gain is ill defined, and training a decision tree is ill posed. However, Dollár and Zitnick observed that contour labels $\mathbf{y} \in Y$ have structure and therefore can be mapped to a discrete label space $C = \{1, ..., k\}$. By mapping similar elements in $\mathbf{y} \in Y$ to the same discrete labels $c \in C$, *approximate* information gain can be calculated across C, and normal training procedures can be used to train decision trees [DZ13].

To find a discrete label set C and an accompanying mapping $Y \rightarrow C$, similarity between individual labels $\mathbf{y} \in Y$ must be measured. Similarity between labels presents computational challenge since comparing two label patches results in $\binom{8\cdot 8\cdot 8}{2} = 130,816$ difference pairs. To avoid this complexity, we map Y to an intermediate space, Z, where similarity can be calculated efficiently. Intermediate label space Z is found by down-sampling Y to m samples

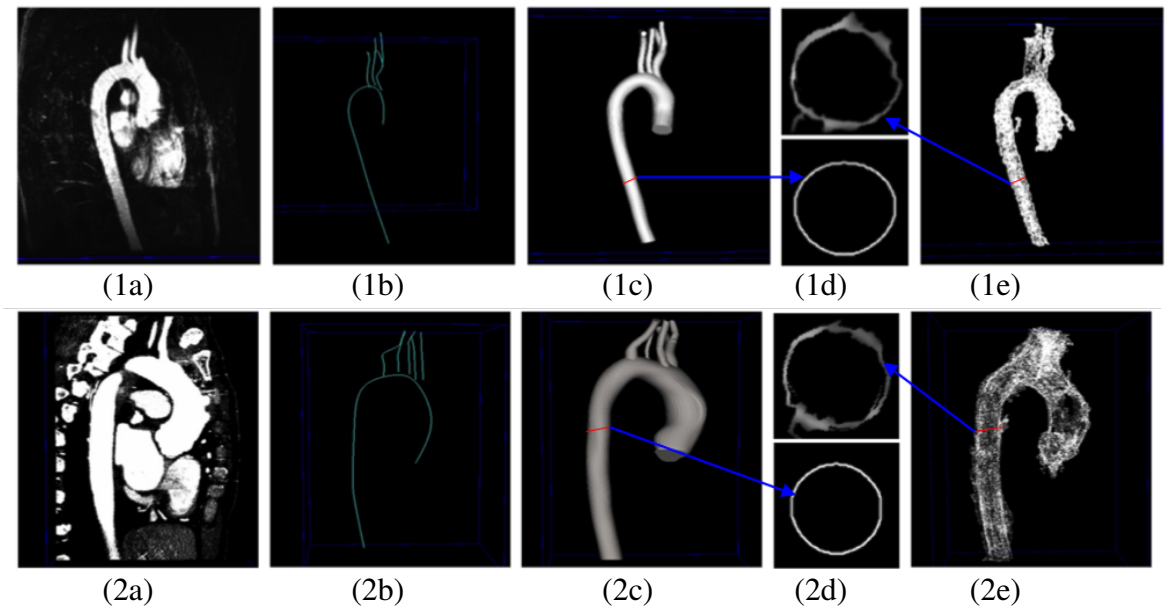


Figure 2.4. Illustration of edge detection results. Columns from left to right: (a) Image data, (b) center line paths, (c) 3D ground truth, (d) vessel cross section (top: our result, bottom ground truth), (e) our result. Row 1 depicts an MR result, row 2 depicts a CT result.

then computing Principal Component Analysis (PCA) and keeping only the λ strongest components. This results in a parameterized mapping $\Pi_\phi : Y \to Z$ where $\phi = \{m, \lambda\}$. Using similarity calculated in Z, mapping $Y \to C$ becomes a straight forward calculation. Figure 2.2 shows an example of structured labels mapped to discrete label sets. During training, decision trees are de-correlated by randomizing sample locations, randomly disregarding 50% of the features, and selecting a random down-sampled set when mapping $\Pi_\phi : Y \to Z$.

Since each tree classifies an entire neighborhood, we performed edge classification for patches centered at odd rows, odd columns, and odd aisles. Every voxel location receives multiple votes from each tree, $t_i \in T$ where $T = (t_1, t_2, t_3, ...t_N)$. With $8 \times 8 \times 8$ label patches, each voxel receives $\approx 8^3 \cdot ||T||/8$ votes, in our experiments we set $||T|| = 8$. Votes were de-correlated by aggregating results from alternating trees at each location.

2.4 Experimentation and Results

We evaluated our method on a data set consisting of 21 MRI and 17 CT volumes for a total 38 volumes that include part or both of the thoracic and abdominal regions. 30 of the volumes in our dataset were provided to us from the vascular model repository (http://www.vascularmodel.com) the remaining eight were collected from local collaborators. Image volumes vary in resolution from $512 \times 512 \times 512$ to $512 \times 512 \times 64$. All volumes were captured from individual patients for clinically indicated purposes then expertly annotated with ground truth and verified by a cardiologist. Ground truth annotations include all visible arterial vessels except pulmonary and coronary arteries and each volume was annotated specifically for CFD simulation. For experimentation, our data set was divided into sets for training and testing. Our classifiers were trained on 29 randomly selected volumes, and the remaining 9 were used for testing, both sets include approximately 50% MRI and CT images.

For experimentation, we trained our classifiers using a combination of C++ and Matlab implementations. Appearance feature extraction and manipulation was performed using the Insight Toolkit (ITK) [YAL+02], and classifiers were trained using Piotr Dollár's Matlab toolbox (http://vision.ucsd.edu/~pdollar/toolbox/doc).

We performed two sets of experiments to examine different aspects of our algorithm. In our first set of experiments, we compared our method against a baseline structured forests model and two PBT forests models, one trained using our feature and sampling methodologies and another trained with baseline features and sampling. In our baseline models, positive samples included patches where any edge voxel is present and negative samples were collected from any negative region without restriction. We selected baseline features to closely mirror features used in natural images, these are: lumen intensity, gradient magnitude and oriented gradients. Oriented gradients were computed by projecting $\nabla F(x,y,z) = (\frac{df}{dx}, \frac{df}{dy}, \frac{df}{dz})$ onto surface of an approximated unit sphere. In addition, we collected results of commonly used

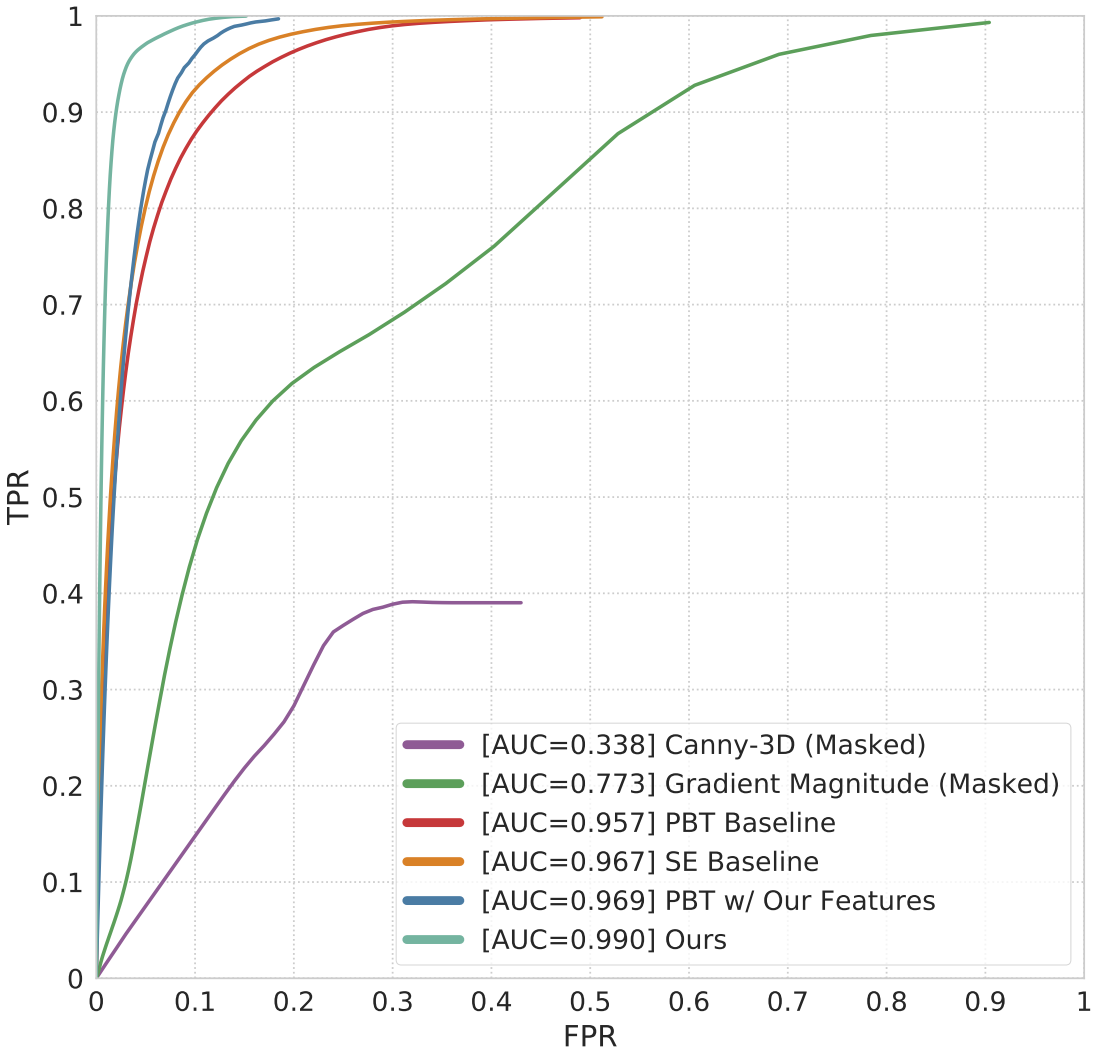


Figure 2.5. Classifier receiver operating curves comparing our method, PBT edge detectors and commonly used edge detectors.

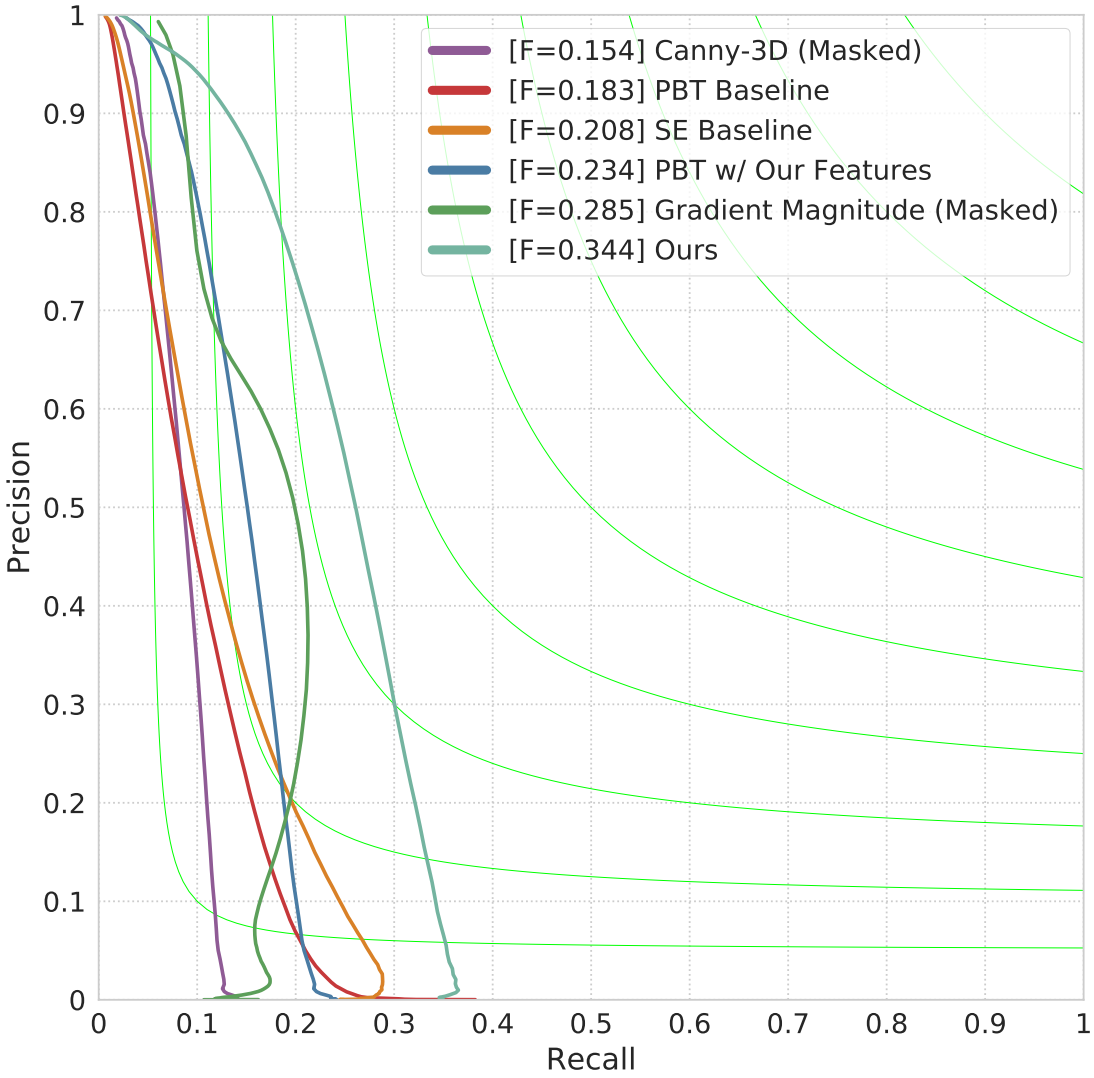


Figure 2.6. Classifier precision vs recall curves comparing performance of our method, PBT edge detectors and commonly used edge detectors.

edge detection methods, gradient magnitude and Canny edge response.

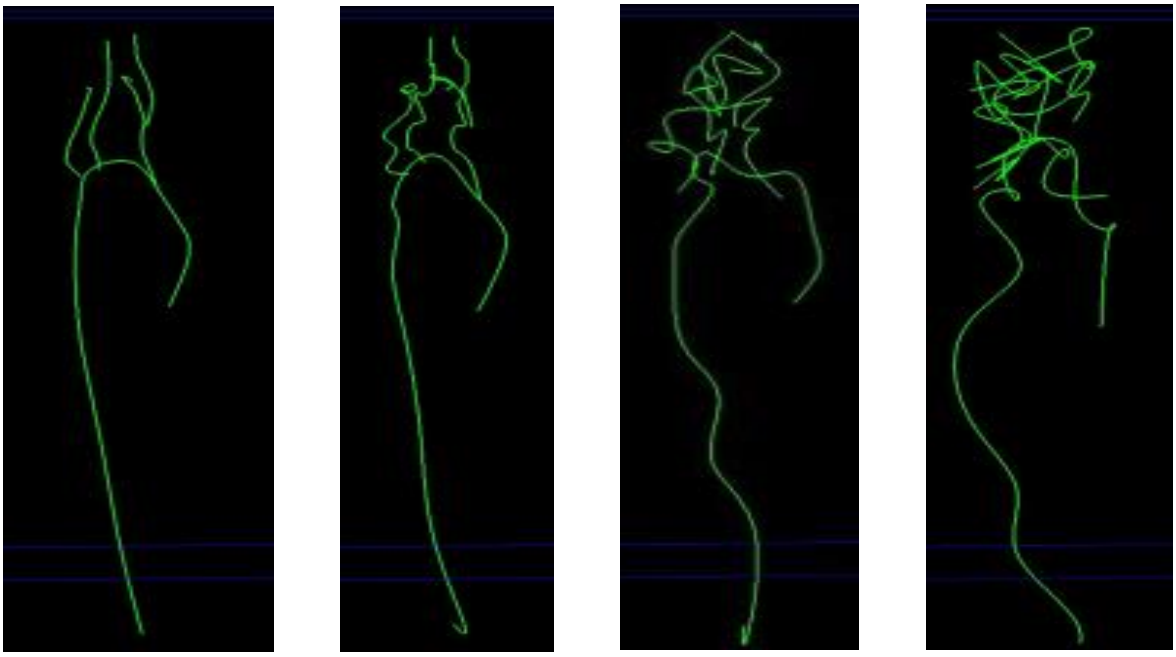


Figure 2.7. Example illustration of path-lines where control points were perturbed with increasing levels of additive noise. Depicted uniform additive noise levels (from left to right): $\pm$0mm, $\pm$1mm, $\pm$10mm and $\pm$25mm. During experimentation noise levels were perturbed with uniform additive between $\pm$0mm and $\pm$50mm

In our second set of experiments, we examined our method's robustness to error in user-generated center-line annotations. For each test volume, the center-line control points were varied by additive uniform noise, center-lines were re-interpolated, and new edge maps were computed. This process was performed at increasing noise levels from $U(-.5\text{mm}, .5\text{mm})$ to $U(-50\text{mm}, 50\text{mm})$, Figure 2.7 shows select examples of the result. Performance statistics were calculated by comparing edge responses against ground truth annotations and hit, fall out and positive predictive voxel rates were calculated by tallying counts for each image and combining them to obtain rates for the entire test set. For our method, PBT and gradient magnitude, performance curves were generated by thresholding edge responses at different values. Canny performance curves were generated by sampling different combinations of input parameters. For each performance curve, area-under-the-curve

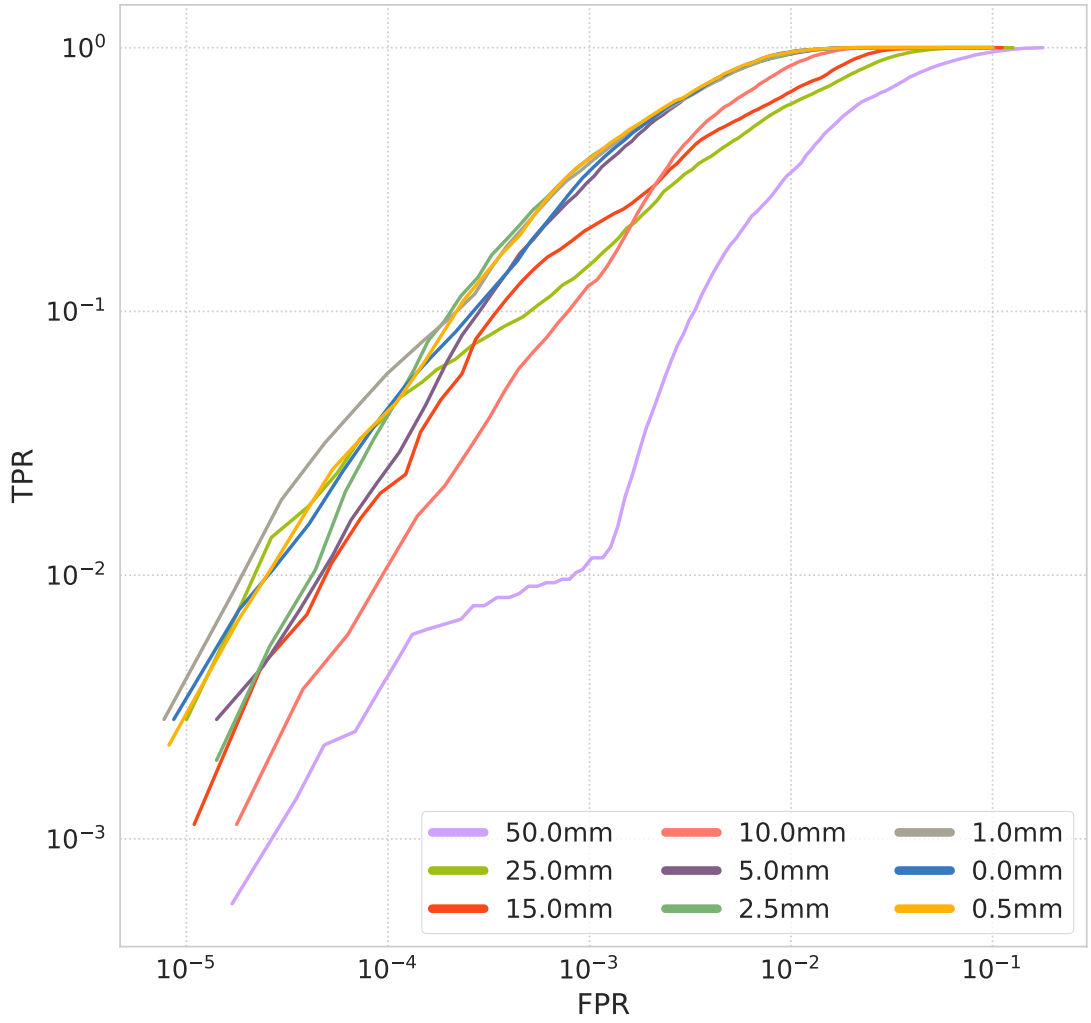


Figure 2.8. Receiver operating curves of SE-3D after after introducing center-line error through uniform additive noise on control points.

(AUC) and top F-Measure scores were recorded and are included in our results. To fairly compare conventional methods against learned edge classifiers, only voxels within a 15 voxel radius of ground truth vessel walls were considered in performance calculations for Canny response and gradient magnitude.

Quantitative results for the first set of experiments are displayed in Figures 2.5 and 2.6. This figure shows that our method out-performs similarly trained PBT classifiers and out-performs commonly used edge detectors by a wide margin. We see both increased

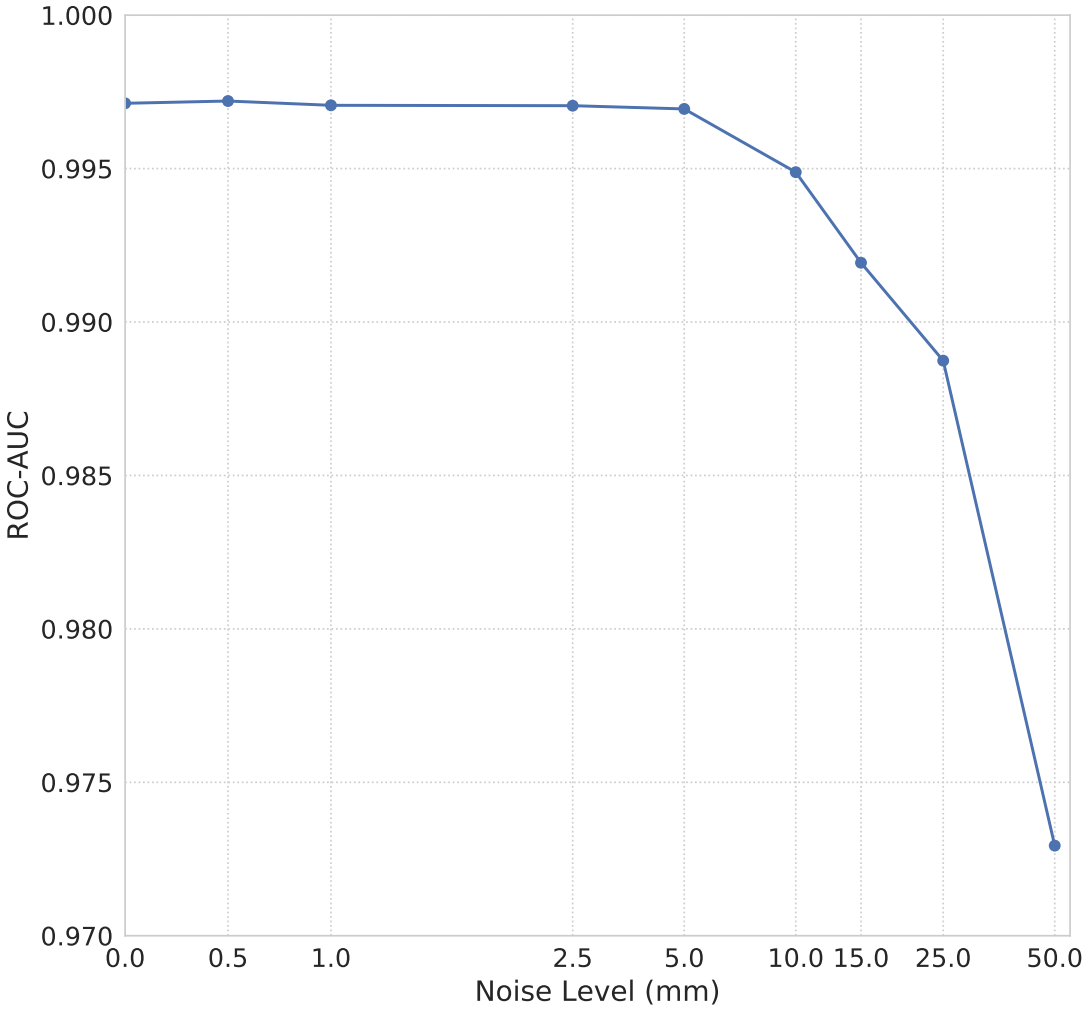


Figure 2.9. ROC AUC of the SE-3D classifiers with various noise levels introduced into the pathline through uniform additive noise on control points.

accuracy and improved precision when utilizing structured output labels and domain specific features. Figures 2.5 and 2.6 further indicate that our features and sampling methodology boost accuracy for other learned edge detectors. Figure 2.4 shows qualitative examples of classifier's results. Results of the second experiment appear in Figure 2.8. These curves suggest that our method is very robust to inaccurate annotation. We see only small a small drop in performance for noise levels up to $U(-2.5\text{mm}, 2.5\text{mm})$, and our method continues to achieve reasonable performance even with high noise introduced to the pathline.

2.5 Conclusion

We introduced a novel method for edge classification in medical imaging with domain specific features capable of effectively capturing edge contours across two imaging modalities. In addition, we incorporate an atlas prior to edge classification and increase performance. We show that to obtain top accuracy and edge localization, it is important to constrain positive and negative sampling regions to areas of interest. Our experiments indicate that our method out performs other edge detectors and show that it is robust to *a prior* error. Given that many medical imaging techniques are built upon edge fields, we feel that this approach has significant potential to a variety of applications.

Acknowledgements

Chapter 2, in part, is a reprint of the material as it appears in Jameson Merkow, Zhuowen Tu, David Kriegman, and Alison Marsden. International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2015. The dissertation author was the primary author of this paper.

Chapter 3

Convolutional Neural Networks for Dense Volume to Volume Labeling

3.1 Introduction

Imaging technologies such as Computed Tomography (CT) and Magnetic Resonance (MR) produce a vast amount of volumetric data that radiologists and other medical professionals are required to analyze and evaluate. For clinicians and researchers alike, computer assisted diagnostic and detection (CAD) systems produce valuable information often required for comprehensive review of this data. Medical image segmentation plays a vital role in many CAD systems by providing detailed annotation and labeling, which in turn allows a more exhaustive analysis to be carried out. We introduce two methods for automatic volume-to-volume labeling, which classify each voxel of a medical image volume input into a finite number of categories.

In most neurological studies, annotation and extraction of brain tissue is the first step to a more sophisticated analysis. For example, brain volumetry [GDS13] and surface reconstructions [TRN$^+$06, FSTD99] require accurate and precise skull stripped scans. Furthermore, the shape of the neurological structures such as gyrus and sulcus provide valuable information in diagnosis of Alzheimer's disease [TMW$^+$01] and other neurological pathologies. Accurate brain boundary localization can aid in the characterization of these structures and subsequent

disease diagnosis.

In addition, medical volume segmentation plays a major role in patient-specific cardiovascular simulation, which, has emerged as an increasingly necessary method to non-invasively obtain detailed hemodynamic data. This data is vital to developing new surgical procedures [EMHMoCHAMI15] and linking hemodynamic factors with clinical outcomes and disease progression [Mar14]. Construction of 3D cardiovascular model from imaging data is a necessary but time consuming precursor to these types of simulations.

Automatic volumetric labeling is a challenging task, given the large anatomic variability and the computational complexity required to process 3D medical data. A popular method to mitigate cost is to repeat a simple core operation on dense, overlapping windows. These "sliding-window" approaches and patch-centric frameworks distribute the complexity into smaller, more manageable operations but limit the long-range modeling capabilities, high-order correlations, and volume-wide information of the underlying classifiers. Computational cost of the sliding window model is still steep, a cost which grows exponentially in volumetric labeling, highlighting the need to move past patch-centric methodologies for volume-to-volume labeling.

Recently, there have been an influx of approaches using deep CNNs aiming to tackle image-to-image prediction. Fully convolutional neural networks (FCN), introduced by [LSD15], represent a proof-of-concept for simultaneous performance and full image segmentation/labeling. [LSD15] modified the VGGNet architecture [CSVZ14] by adding element-wise summations to link coarse predictions to layers with finer strides. [XT15] applied another approach to image-to-image object boundary detection creating the popular Holistically-Nested Edge Detector (HED), and tackled two key issues of this long-standing vision problem: (1) holistic image training and prediction; and (2) multi-scale feature learning. HED, guided by deep supervision [LXG+15], merges boundary predictions from multiple resolutions to resolve ambiguities. However, CNN architectures such as FCN and HED are

fundamentally limited by their fine-to-coarse structure. [RFB15] extended FCN by replacing summation layers concatenation and convolution layers to produce fine-to-fine predictions, but made no attempt to learn multi-resolution features.

Early approaches to volume-to-volume prediction treat each slice of the volume as a 2D image, and use post-processing to recombine the results back into 3D. This approach lacks the capability to model volumetric features and therefore has fundamental context limitations. Some patch-to-patch methodologies have been proposed; [MTKM15] outlines one such strategy, where appearance features and *a-priori* information are used as input to a 3D extension of the popular structured forest classifier [DZ15] to make cardiovascular boundary predictions. That work side-steps the inefficiency of patch-centric classifiers by employing a specialized sampling scheme and limiting prediction to a narrow set of structures. Other volumetric approaches using CNNs have been attempted, such as [ZLG$^+$15], but these architectures continue to incur a high computational cost which make them unsuitable for volume-to-volume tasks.

In addition to the computational challenges inherent in volume-to-volume prediction, medical volume labeling requires greater precision than typical natural image tasks due to the prevalence of small, yet important structures that need to be detected. Unlike natural images, where small structures can generally be ignored, localizing small abnormalities and subtle anatomical changes is crucial for diagnosis and patient care in applications that include cancer tumor detection, coronary atherosclerosis, neuroscience, and others.

Here, we introduce two volume-to-volume frameworks that emphasize efficiency, precision and a fine-to-fine strategy coupled with nested multi-level and multi-scale learning. First, we describe our extension of the popular 2D-CNN HED [XT15] for volumetric labeling then build upon this work and introduce a new architecture, I2I-3D, aimed at tackling three critical issues in precise volume-to-volume labeling: (1) efficient volumetric labeling of medical data using 3D, volume-to-volume CNN architectures, (2) precise fine-to-fine and

volume-to-volume labeling at input resolution, (3) nested multi-level, multi-scale feature learning. We evaluate our approach on two challenging medical imaging task across two publicly available datasets. We compare against multiple baselines including 2D and 3D CNN strategies and achieve state-of-the-art performance on vessel wall segmentation, brain boundary detection and skull stripping.

3.2 Dense Volume to Volume Labeling

3.2.1 Existing Pixel Level Prediction

In this section, we describe, in detail, the foundations of our approach to dense volume-to-volume labeling. We begin by discussing related CNN approaches, in particular those that perform dense pixel predictions on 2D images. Many CNNs that perform dense pixel-level predictions derive from VGGNet [CSVZ14]. Each unit in VGGNet contains 2-3 successive convolution layers followed by a pooling layer. Convolution layers produce robust features, and pooling layers distill these features into a smaller spatial extent, allowing diverse, long-range responses to be learned later in the network. As these units are repeated, features are pooled and processed at a lower resolution to form more abstract features with a larger spatial footprint. This architecture has proven powerful in a number of classification tasks due to its rich set of features across multiple scales but it creates two critical challenges for image-to-image or volume-to-volume prediction: (1) the most powerful representations, in terms of complexity and depth, also have the lowest resolution and (2) coarse resolution information cannot be used at finer resolutions, therefore no coarse level guidance exists.

Some approaches aim to mitigate these limitations. [LSD15] adapted the VGGNet architecture by adding skip connections which link coarse resolutions to finer ones. [XT15] proposed an alternative adaptation, where multi-resolution outputs are fused through weighted aggregation, producing a state-of-the-art boundary detector. In yet another adaptation, [HCH$^+$16] added upsampled skip connections to the HED architecture to increase recall of

higher resolution side-outputs. These approaches avoid the fine-to-coarse limitations but they do not tackle these limitations directly. Furthermore, [HAGM14a] show that directly merging low resolution features with finer ones is sub-optimal. An inspection of the outputs of these networks reveals coarse predictions that leave major ambiguities which can only be delineated during post processing. HED produces many thick, orphan edges and requires post-processing with non-maximum suppression and morphological thinning to increase resolution [XT15]. Outputs from FCN have been refined via conditional random fields to increase precision of its coarse predictions as detailed by [ZJRP$^+$15]. Recently, the UNet architecture used an end-to-end approach to neuronal segmentation [RFB15]. UNet's architecture merges resolutions and applies a penalty for poor localization, but it does not directly learn nested multi-scale features, furthermore, its layer density make it too inefficient for volume-to-volume predictions. Some of these limitations of UNet were addressed in a follow up work UNet-3D [CALR17], however, similar to other 3D networks such as [TSR$^+$15], [CALR17] does not, directly, account for multi-scale features. All of these approaches, therefore, have fundamental limitations to precise end-to-end, volume-to-volume prediction in many application of medical image analysis.

3.2.2 From Fine-to-Coarse to Fine-to-Fine

We start by examining different multi-resolution merging configurations: downstream merging, upstream merging, nested multi-level merging and multi-level merging with directed multi-scale learning. Figure 3.3 illustrates of these four strategies. Downstream-merging starts by producing multiple predictions at different resolutions then, as the name suggests, merges the multi-resolution predictions downstream, typically through aggregation. As depicted in Figure 3.3(a), in this type of architecture processing moves strictly fine-to-coarse and the burden of fine localization is placed solely on early layers. An alternative strategy, illustrated in Figure 3.3(b), is to merge resolutions upstream by pulling deeper/lower resolution predictions

directly into higher resolution side-outputs. This upstream merge strategy aims to alleviate the burden on early stage side-outputs by providing coarse prediction information, and in-turn the higher resolutions features are freed-up to resolve the ambiguities left by the coarse predictions. The final strategies appears in Figure 3.3(c) and 3.3(d). In contrast to the other two approaches, these strategies integrate multiple resolutions at the feature level rather than at the prediction level. Architectures following the strategy depicted in Figure 3.3(c) integrate multi-level features but do not use any sort of multi-scale learning, discouragin multi-level representation since their in no cost for failing to do so. Our strategy, shown in Figure 3.3(d), combines low and high resolution features through a resolution 'mixing' procedure, and rewards mult-scale integration at each stage with multi-scale loss thereby directly enforcing multi-resolution merging. We discuss these strategies in greater detail in subsequent sections.

3.2.3 Nested Multi-level Learning

Our framework tackles three crucial aspects of fine-to-fine, volume-to-volume prediction: (1) efficient labeling of 3D medical volumes, (2) holistic volumetric training and prediction of precise voxel-level labels (3) nested multi-scale, and multi-level feature learning. By making high resolution predictions at deeper layers, we enable the network to produce predictions that benefit from network depth, maintain higher precision and learn the complex interaction between multiple scales.

I2I builds upon previous works in holistic prediction by introducing nested coarse-to-fine feature learning which turns the typical fine-to-coarse network into a precise fine-to-fine classifier. The fine-to-coarse path of I2I resembles popular architectures and contains several cascaded units each consisting of two to three convolution layers followed by a pooling layer. Adding side-outputs and weighted aggregation to this fine-to-coarse network results in our HED-3D architecture, which, efficiently produces accurate results but, unsurprisingly, the labels lack precision.

To create our I2I framework, we append an additional coarse-to-fine structure that systematically merges fine and coarse features. Each stage in our coarse-to-fine path carries out representational and spatial mixing, combining coarse and finer resolution responses into new higher resolution predictions. We use deep supervision [LXG+15] at each stage to enforce a multi-resolution loss function, rewarding integration of coarse representations into finer features. Finer resolution features benefit from abstract features and coarse level guidance, culminating in an output layer with the finest resolution, as well as the most predictive power. The favorable characteristics of these underlying techniques manifest in I2I being accurate, precise and computationally efficient.

3.2.4 Formulation

We denote our input training set of N volumes by $S = \{(X_n, Y_n),\, n = 1, \ldots, N\}$, where sample $X_n = \{x_j^{(n)}, j = 1, \ldots, |X_n|\}$ denotes the raw input volume which is paired with a corresponding ground truth label map $Y_n = \{y_j^{(n)}, j = 1, \ldots, |Y_n|\}, y_j^{(n)} \in \{1, .., K\}$ where K is the total number of semantic classes. Our examples use only two classes ($K = 2$), however in this formulation, we define the generic loss formulation for K classes. As we consider volumes independently, n is dropped for simplicity. Our goal is to learn network parameters, $\mathbf{W}$, that produce M outputs, each containing voxel labels at different resolutions. Each output has a resolution of $\frac{1}{2^{m-1}}$ of the input resolution. All outputs use a classifier whose weights are denoted $\mathbf{w} = (\mathbf{w}^{(1)}, \ldots, \mathbf{w}^{(M)})$. Loss for each of these outputs is defined as:

$$\mathscr{L}_{\text{out}}(\mathbf{W}, \mathbf{w}) = \sum_{m=1}^{M} \ell_{\text{out}}^{(m)}(\mathbf{W}, \mathbf{w}^{(m)}), \tag{3.1}$$

where ℓ_{out} denotes the volume-level loss function. Loss is computed over all voxels in a training volume X and label map Y. Specifically, we define the following cross-entropy loss function used in Eqn. (3.1):

$$\ell_{\text{out}}^{(m)}(\mathbf{W}, \mathbf{w}^{(m)}) = -\sum_{k} \sum_{j \in Y_k} \log \Pr(y_j = k | X; \mathbf{W}, \mathbf{w}^{(m)}), \tag{3.2}$$

where Y_k denotes the voxel truth label sets for the k^{th} class. $\Pr(y_j = k|X; \mathbf{W}, \mathbf{w}^{(m)}) = \sigma(a_j^{(m)}) \in [0,1]$ is computed using sigmoid function $\sigma(.)$ on the activation value at voxel j. We obtain predictions $\hat{Y}_{\text{out}}^{(m)} = \sigma(\hat{A}_{\text{out}}^{(m)})$, where $\hat{A}_{\text{out}}^{(m)} \equiv \{a_j^{(m)},\ j = 1, \ldots, |Y|\}$ is the output of layer m. Putting everything together, we minimize the following objective function via standard stochastic gradient descent:

$$(\hat{\mathbf{W}}, \hat{\mathbf{w}}) = \underset{\mathbf{W}, \mathbf{w}}{\text{argmin}}(\mathscr{L}_{\text{out}}(\mathbf{W}, \mathbf{w})), \tag{3.3}$$

During evaluation, given image X, we obtain label map predictions from the output layers via the following operation: $\hat{Y}_{\text{top}} = \text{Net}(X, (\mathbf{W}, \mathbf{w}))$, where $\text{Net}(\cdot)$ denotes the label maps produced by one of our networks.

3.3 Network Architectures

In this section, we discuss the architectures used for volume-to-volume prediction. Figures 3.1 and 3.2 depicts our two architectures, HED-3D and I2I-3D.

3.3.1 HED-3D

With 16 single stride convolution layers and multiple stages at different resolutions, VGGNet [CSVZ14] provides a great starting point for accurate volume-to-volume labeling. Our fine-to-coarse classifiers build upon VGGNet's structure by adapting it to 3D and making domain specific modifications. First, we reduce the number of filters in the first two layers (conv1_2 and conv1_2) to 32 from 64. Medical volumes contain less texture and other low-level complex cues, therefore fewer low-level features are required. We also remove all fully connected layers and the fifth pooling layer (pool5). We make two optional modifications: 1)

the type of pooling used is modified (max or average), and 2) we change the total number of stages (resolutions).

Our vascular boundary detector uses average pooling with either four (I2I-3D) or five stages (HED-3D); all of the skull stripping and brain boundary classifiers use max pooling and four stages. Modifications for each application are summarized in Table 3.1. To build our HED-3D classifier, we add side-outputs and deep supervision just prior to each pooling layer and a fusion layer which merges each side output with learned weighted aggregation. HED-3D uses downstream merging as depicted in Figure 3.3(a).

3.3.2 Densely Connected HED-3D

We also introduce an alternative 3D classifier that merges resolutions upstream by linking deeper side outputs directly to shallower ones with skip connections. This network uses a simple structure that merge deeper, lower resolution responses with weaker, higher resolution side outputs. This method differs from HED in how in that each side output is merged with all those at lower resolution. This architecture was recently proposed in [HCH+16]; we follow design modifications outlined there, applying them to our HED-3D architecture. We report results of this classifier as an alternative strategy for multi-resolution merging. We refer to this architecture style as densely connected HED-3D, or dHED-3D for short. This strategy is depicted in Figure 3.3(b).

3.3.3 I2I-3D

I2I-3D uses a coarse-to-fine structure to systematically integrate multiple resolutions and learn nested multi-scale features. This structure follows an inverse pattern of the fine-to-coarse structure described in Section 3.3.1. Upsample layers replace pooling layers and higher resolutions appear later in the network. The coarse-to-fine path is divided into distinct stages, each of which take inputs from multiple resolutions. In each stage, a series

of specialized convolution layers mix these two inputs. Initially, the two representations are directly mixed, channel by channel, via a 'mixing layer'. Mixing layers perform two operations simultaneously: concatenation and channel pooling via a $1 \times 1 \times 1$ convolution layer. These operations are conceptually similar to 'reduction' layers used by [SLJ+15], but differ in purpose. In addition to maintaining network efficiency, these layers produce an amalgam of features from two resolutions. Feature mixing results are passed to two convolution layers that spatially mix the two streams. Effective multi-resolution mixing is enforced by deep supervision (DSN) at the end of each stage. DSN plays an important role in our architecture, as it forces each stage to merge powerful low resolution features with finer resolution features to minimize loss at each stage. After these multi-resolution features are learned, all lower resolution outputs are removed leaving a single high resolution output. Figure 3.3(d) depicts the merging strategy used by I2I-3D.

3.4 Implementation

Our 3D-CNN, publicly available at https://github.com/jmerkow/I2I implementation is based on the popular *Caffe* library [JSD+14]. Volume manipulation is performed using the ITK library [JMI15].

3.4.1 Data Preparation

Prior to training and testing, each volume is cropped into $96 \times 96 \times 48$ overlapping segments following a volume-wide voxel intensity whitening step. Each training volume segment overlaps its neighbors by an eighth of the segment size ($12 \times 12 \times 8$ voxels) resulting in approximately 37.5% overlap during training. To avoid class imbalances, only volumes with over 0.125% positively labels voxels were trained on (approx. 500 of $442,368$ voxels).

3.4.2 Network training

We begin by describing common training procedures among our architectures. All of our network's fine-to-coarse weights are initialized from pretrained weights. Our pretrained weights were generated by training a generic fine-to-coarse architecture for a segmentation task. This network was initialized with using Xavier random initialization and trained with a high learning ($1e$-7) for a fixed number of epochs (25). Hyper parameters for this process were not tuned, the highest learning rate which did not diverge for at least 25 epochs was selected. This process simply generates meaningful weights for initializing networks when training for other tasks. All other hyper-parameters were optimized based on performance on a validation set.

All data is shuffled after every epoch during training. When training to predict vascular boundaries, segments belonging to the same volume were shuffled followed by a randomization of the order of volumes; all segments from an individual volume were trained on prior to moving onto the next volume in the sequence. After each epoch the intra-volume segments were reshuffled as was the order of the volumes. On brain and skull data, segments from all volumes were shuffled together and randomized again after every epoch. Given that all images in our brain and skull datasets were a single modality, intra-volume shuffling did not improve results. The network-wide learning rate was reduced using the step policy:

$$\mathrm{LR}(\mathrm{epoch}) = \mathrm{base}_{LR} \cdot \gamma^{\lfloor \frac{\mathrm{epoch}}{\mathrm{step}} \rfloor} \tag{3.4}$$

Update optimization was calculated using standard stochastic gradient. Hyper-parameters used during training are summarized in Table 3.1.

I2I-3D is trained in three phases. In first phase, only the fine-to-coarse path of I2I-3D is trained, hyper-parameters are identical to those used for our HED-3D classifier (as summarized in Table 3.1). Both the first phase I2I-3D and HED-3D are trained with a side

Table 3.1. Table of hyper-parameters and network configurations we use to train our architectures on different tasks. Parameters were selected from a sweep based on validation set performance criteria. HED-3D hyper-parameters were used to train the fine-to-coarse path of our I2I-3D framework. I2I-3D values listed here refer to those used during the second phase of training.

	Architecture	Base LR	Step Size (Epochs)	γ (Gamma)	Weight Decay	Batch Size	Pooling Type	Resolutions
Vascular	UNet-3D	$1e$-8	18	0.1	$1e$-4	1	MAX	4
	HED-3D	$1e$-8	18	0.32	$1e$-4	1	AVE	5
	dHED-3D	$1e$-8	18	0.32	$1e$-4	1	MAX	4
	I2I-3D	$1e$-7	6	0.1	$1e$-4	1	AVE	4
Skull Stripping	Kleesiek et al.	$1e$-8	20	0.32	$1e$-4	1	MAX	2
	UNet-3D	$1e$-8	20	0.1	$1e$-4	1	MAX	4
	HED-3D	$1e$-8	20	0.32	$1e$-4	1	MAX	4
	dHED-3D	$1e$-8	20	0.32	$1e$-4	1	MAX	4
	I2I-3D	$1e$-7	6	0.1	$1e$-4	1	MAX	4
Brain Boundary	Kleesiek et al.	$1e$-7	20	0.32	$1e$-4	1	MAX	2
	UNet-3D	$1e$-7	20	0.1	$1e$-4	1	MAX	4
	HED-3D	$1e$-8	20	0.32	$1e$-4	1	MAX	4
	dHED-3D	$1e$-8	20	0.32	$1e$-4	1	MAX	4
	I2I-3D	$1e$-7	6	0.1	$1e$-4	1	MAX	4

output and loss layer at each scale (as described in 3.1), and an additional output with loss at the fusion layer; an example illustration of this network (with 5 scales) is depicted in Figure 3.1. The fusion layer of HED-3D is used during evaluation, as it provided the best results. During training, we drop balanced cross entropy loss as used in [XT15]. Given that we train on volumetric segments as described in Section 3.4.1, this cost function was no longer necessary. Training a powerful fine-to-coarse classier is critical to producing an accurate and precise fine-to-fine network; this ensures meaningful representations at each resolution, creating a strong starting point for multi-scale integration. During the second phase, we attach the coarse-to-fine path and initialize each layer as outlined in Section 3.4.3. We train phase two for fewer epochs per step with a higher learning rate. During this training phase, weights in the fine-to-coarse path are fixed by setting their learning rate multipliers such that they update 100 times slower than that of the rest of the network. Given that these additional weights are freshly initialized, we increase the learning rate to compensate. These learning rates force the coarse-to-fine path integrate their multi-resolution inputs to minimize loss rather than update fine-to-coarse features. During the final phase, we lower the network-wide learning rate to a tenth of the previous rate, and return all learning rate multipliers to their original values to train for an additional 3-4 epochs.

3.4.3 Weight Initialization

Since the secondary pathway aims to incorporate complex, low resolution information into higher resolution responses, it is important that this information is not degraded when incorporated into higher resolutions. We preserve these representations through careful initialization of the coarse-to-fine layers before training phase 2. Each stage is initialized such that their output is an identity mapping of the higher resolution inputs. Specifically, we initialize the parameter weights of each mixing layer to one at locations corresponding to finer resolution inputs and all others are set to zero. The two convolution layers are initialized with

a one in the spatial center along the channel diagonals and otherwise zero. This initialization strategy forces each stage to output only higher resolution features, the weights in the fine-to-coarse path are fixed so, in order to reduce loss at each resolution, coarse representations must be integrated. In this way, we leverage the power of the lower resolution features to make accurate, though imprecise, predictions and allow multi-resolution mixing to occur naturally. With random initialization of these layers, the signal from the fine-to-coarse layers become corrupt and the network cannot integrate multi-scale features efficiently. We found through experimentation that when these layers are initialized randomly, the full I2I network performs sub-optimally.

3.5 Experimentation and Results

In this section, we outline our experimental setup and report performance of the two algorithms, HED-3D and I2I-3D and compare them to multiple baselines. On all tasks we compare our method to UNet-3D [CALR17], HED [XT15], and dHED-3D as described in Section 3.3.2. On our brain-related tasks, we compare to two additional baselines, the All-CNN architecture proposed by [KUH$^+$16] and the popular ROBEX [ILTT11] classifier. On vascular boundary detection, we compare against SE-3D [MTKM15] as well as Canny-3D.

During experimentation, each network was trained as outlined in Section 3.4. The 2D classifier, HED, was trained on slices taken from each volume in the training set following the steps and hyper-parameters outlined by [XT15]. For evaluation, HED label maps were computed on sequential 2D slices and then combined back into a 3D volume. All CNN classifiers were trained with with the hyper-parameters that resulted in the best performance on a validation set during a parameter sweep. Hyper-parameter values are summarized in Table 3.1.

3.5.1 Datasets

First, we evaluate our frameworks on the LONI Probabilistic Brain Atlas, commonly called LPBA40, dataset performing two separate tasks: skull stripping and brain-boundary detection. Atlas information plays important roles for visualization, interpretation and analysis of brain data.

LPBA40 is a publicly available dataset of 40 MRI volumes, each accompanied with a manually annotated brain mask used for skull stripping. Examples of data from this dataset can be found in Figure 3.4. Brain boundaries, for both training and testing, were obtained from brain masks through morphological manipulation. This dataset was randomly split into 25 training volumes, 5 validation volumes, and the remaining 10 were used for testing.

Second, we evaluate our approach on blood vessel boundary detection an extension of the dataset used by [MTKM15], that has been expanded to incorporated a wider variety of physiologies and anatomical regions. The dataset was expanded from the oringal 38 used by [MTKM15] with 65 additional volumes to form a dataset of 93 volumes containing, roughly, an equal number of MR and CT volumes. Though many approaches train separate networks for each modality, we found that training a single network without regard to modality (after appropriate preprocessing steps) to be more efficient. An example of data from this dataset can be found in Figure 3.5. The dataset contains various arterial vessel types, but only one structure is annotated per volume. Each volume was captured from individual patients with a wide variety of pathologies including: aneurysms, stenoses, peripheral artery disease, congenital heart disease, and dissection as well as normal physiologies. Each volume is accompanied with an expertly annotated 3D model, built for computational blood flow simulation in the open source SimVascular package [UWM+13]. All volumes from this dataset are publicly available[1] at the vascular model repository [WOJ13]. During experimentation, these volumes were split into three sets: training, validation, and testing each containing 67, 7 and 19

[1] http://www.vascularmodel.com

volumes respectively. Each set contains approximately the same ratio of CT and MR volumes (~50% of each). Volumes in this dataset are under-annotated and a mask was introduced during evaluation so that only voxels inside annotated vessels and those within 20 voxels of the vessel wall were considered. Phase 1 and our HED-3D networks were initialized with pre-trained weights learned from segmenting entire vessels.

3.5.2 Metrics

Skull Stripping

For skull-stripping evaluation, three standard metrics are used: precision-recall curves, DICE score, and F-measure. To fully evaluate segmentation map quality, we use the F-measure metric which is defined as follows:

$$F_\beta = \frac{(1+\beta^2)\,\text{precision} \times \text{recall}}{\beta^2\,\text{precision} + \text{recall}}. \tag{3.5}$$

We set $\beta^2 = 0.3$, as suggested by other works, which stresses importance of the precision [HCH+16].

Boundary Detection

On boundary detection tasks, we evaluate using performance benchmarks introduced by [MFTM01] which are standard protocols for evaluating boundary contours in natural images. Voxel overlap is not well-suited for boundary detection evaluation as it fails to account for any localization error in boundary prediction and over-penalizes usable boundaries that do not perfectly overlap with ground truth boundaries. The BSDS metrics use correspondence to match true boundaries with predicted boundaries. Corresponding voxels contribute to true positive counts, while unmatched voxels contribute to fall-out and miss rates. Our code that extends these metrics to 3D is publicly available at https://github.com/jmerkow/segbenchpy. We report four performance measures: dataset-fixed threshold F measure (ODS), best per-

image threshold F measure (OIS), average precision (AP) and precision-recall curves.

On all datasets and tasks, metrics were collected through a threshold sweep of the continuous label map outputs; hits, misses and false alarms are counted at every threshold on each volume. Using these counts, we generated precision-recall curves by summing counts at each threshold across the entire dataset. ODS, DICE, F_β scores were calculated using these dataset-wide aggregated counts, but OIS scores were found by averaging the best F score per volume. 3D non-maximal suppression was performed on all boundary label maps prior to thresholding.

3.5.3 BSDS Results

Table 3.2. Summary statistics of BSDS results with and without non maximal suppression. I2I-2D produces significantly better performance when evaluated against other state-of-the-art algorithms without post-processing. We produce more precise pixel level responses and display competitive performance with post processed responses.

	ODS	OIS	AP
Human	.80	.80	-
With NMS			
Canny	0.600	0.640	0.580
SE-Var [DZ15]	0.746	0.767	0.803
DeepEdge [BST15]	0.753	0.772	0.807
DeepContour [SWW$^+$15]	0.756	0.773	0.797
HED [XT15]	0.782	0.804	0.833
I2I-2D (ours)	0.779	0.797	0.789
Without NMS			
SE-Var [DZ15]	0.589	0.601	0.599
HED [XT15]	0.583	0.596	0.570
I2I-2D (ours)	**0.627**	**0.635**	**0.729**

In addition, to the 3D medical tasks, we validate our approach in 2D to show its efficacy in for precise end-to-end predictions. We show results of our classifier's robust and precise classification without post-processing by comparing the I2I principles in a 2D network for a popular 2D task, boundary detection, against state-of-the-art classifiers in that area.

[MFTM01] is a highly competitive dataset for natural images and has driven object boundary and edge detection since its creation in 2001. BSDS is composed of 400 images with manually labeled ground truth contours. This dataset has a predefined split with 200 training, 100 validation, and 100 test. Recently, [XT15] produced near human level accuracy on this dataset, but this result relies heavily on using non-maximum suppression and morphological thinning. We evaluate our method against the state-of-the-art result with and without post-processing.

We start by loading VGGNet pre-trained weights, but increase the number of training iterations to 20,000, keeping the starting base learning rate at 1e-6, which is decimated after 5000 iterations, all other hyper-parameters remain unchanged. We found that the secondary training stage (supervision on the coarse-to-fine path) provided little benefit for this task, so these outputs were removed, and we trained the entire network with only supervision at the top most output. While training the full network, we increased the learning rate step size so that the the learning rate was reduce by a factor of 10 every 12,000 iterations, and decreased the batch size to 5. We trained for 24,000 iterations then completed training with increased learning rate multipliers in the fine-to-coarse path for only 6000 iterations.

Without post-processing our method significantly out-performs the current state-of-the-art, HED which is clear from Figure 3.7. Qualitative results appear in Figure 3.6 which shows much finer resolution responses from I2I-2D than HED, but maintains near state-of-the-art accuracy and compares favorably to structured forests.

3.5.4 LPBA40 Results

On the LPBA40 dataset, we evaluate our approach on two tasks, brain boundary detection and skull-stripping. Skull stripping networks were trained on the brain-mask ground truth included in the dataset. For brain-boundary detection, new networks were trained on brain boundary voxels, these annotations were derived from the original brain mask ground truth data through morphological operations. Example detection for both skull-stripping and

Table 3.3. Results on brain boundary detection. Summary statistics of our approach and baselines.

	ODS	OIS	AP
HED [XT15]	0.688	0.695	0.574
[KUH$^+$16]	0.423	0.425	0.199
dHED-3D	0.613	0.619	0.448
HED-3D (ours)	0.632	0.639	0.489
UNet-3D [CALR17]	0.680	0.683	0.537
I2I-3D (ours)	**0.714**	**0.717**	**0.600**

brain boundaries appear in Figure 3.4.

Brain Boundary Detection

First, we discuss results from our brain boundary detection evaluation. Figure 3.8 depicts precision/recall curves; a table of summary statistics appears in Table 3.3. We compare our approaches with HED applied to slices, dHED-3D, UNet-3D [CALR17], and [KUH$^+$16]. Here, we omit Canny-3D results as brain-boundary detection requires context information that canny detectors cannot model resulting in especially poor performance.

Interestingly, we notice that HED achieves second best performance on this task. Each volume is relatively registered to a reference skull, and there is little variation along the other two axes. We theorize that the performance boost from 3D features is, therefore, not as advantageous. When evaluating on a more diverse dataset, with variation along all axes, we suspect that we would see HED's performance drop significantly, as we do in vascular boundary detection. Furthermore, we observe in Figure 3.4 that HED boundary predictions are considerably thicker than any of the predictions from the 3D classifiers. It is apparent that HED relies heavily upon non-maximal suppression, where as the 3D classifiers produce higher resolution predictions. We also notice artifacts in predictions from the 2D classifier in the axial and sagittal planes, likely, due to the lack of 3D contextual cues. When comparing UNet-3D to our approach, we see that our classifier consistently out-performs

UNet-3D. Particularly, we notice, from the performance curves in Figure 3.8, that UNet-3D exhibits similar precision/recall trade-offs, but at a consistently lower precision rate, providing evidence for the effectiveness of directly learning nested multi-scale features. [KUH+16] shows poor performance on brain boundary detection, we theorize that the required large-scale contextual information for boundary detection is not captured by this network given its shallow depth and few spatial representations. All in all, I2I-3D significantly outperforms all other classifiers according to all metrics with an increase of approximately 5% in ODS and OIS scores, and roughly 12% in average precision. I2I-3D achieves a particularly large improvement in precision as observed from the shape of the precision/recall curve in Figure 3.8.

Skull Stripping

For the skull stripping task, we compare performance of our approaches with HED applied to slices, dHED-3D, UNet-3D [CALR17], the ROBEX skull stripping application [ILTT11] and the current state of the art, [KUH+16]. Table 3.4 shows summary statistics and precision recall curves appear in 3.8. Again, we observe that our I2I-3D classifier achieves top performance according to all metrics. Our 3D-CNN approaches show a large improvement in performance over the popular ROBEX method [ILTT11] and the All-CNN network in [KUH+16].

Table 3.4. Results on skull stripping. Summary statistics of our approach and baselines.

	DICE	F_β
HED [XT15]	96.18	0.9680
ROBEX [ILTT11]	96.60	0.9698
dHED-3D	97.06	0.9761
[KUH+16]	97.34	0.9775
HED-3D (ours)	97.65	0.9809
UNet-3D [CALR17]	97.78	0.9814
I2I-3D (ours)	**97.85**	**0.9826**

When comparing HED-3D and HED, we find evidence of the benefit of using 3D-CNNs over their 2D counter parts. A comparison of HED-3D and I2I-3D, shows an increase in the DICE and F_β scores, indicating that our classifiers have greater precision without loss of recall. On this task, we notice that UNet-3D and I2I-3D have much closer performance measures, however I2I-3D performance is consistently better across all metrics. We observe from Figure 3.4, as seen in brain boundary detection striking noise artifacts in predictions from the 2D-CNN, HED. We can infer that HED produces these artifact in the axial and sagittal views (columns 4 and 6 from right to left of Figure 3.4) as a result of its inability to capture 3D contextual information. We do not see any such artifacts in any 3D classifiers, but rather smooth predictions across all three dimensions. [KUH$^+$16] performs much better at this task than at brain boundary detection. Skull stripping requires shorter range context than a boundary detection, we theorize that this causes the disparity between performances on skull stripping and brain boundary detection for this classifier. Lastly, we observe that dHED-3D has the worst performance across all 3D-CNNs, indicating that its upstream merging strategy is not only less effective but can also be detrimental.

3.5.5 Vascular Boundary Results

Table 3.5. Results on vascular boundary detection. Summary statistics of our approaches and baselines.

	ODS	OIS	AP
SE-3D [MTKM15]	0.303	0.316	0.149
Canny-3D	0.351	0.545	0.241
HED [XT15]	0.529	0.542	0.182
dHED-3D	0.454	0.463	0.271
HED-3D (ours)	0.515	0.528	0.362
UNet-3D [CALR17]	0.550	0.562	0.386
I2I-3D (ours)	**0.567**	**0.580**	**0.421**

Next, we turn to the more difficult problem of vessel boundary detection. This dataset

is considerably larger than the LPBA40 dataset, and spans multiple imaging modalities. Image data is collected from various anatomical regions and volume orientation is not registered to a reference, making this dataset more challenging than the previous ones. In addition to the CNN baselines, we compare HED-3D and I2I-3D to the popular Canny edge detector, and the current state-of-the-art in vascular boundary detection, SE-3D [MTKM15].

Precision/recall curves appear in Figure 3.10 and summary statistics are shown in Table 3.5. Figure 3.5 depicts example detection of I2I-3D and HED-3D. I2I-3D achieves top performance across all metrics with an ~3% increase in ODS and OIS scores, and ~9% increase in average precision score to its closest competitor (UNet-3D). Interestingly, we notice that HED produces high ODS and OIS scores. However, all 3D methods have higher average precision and we observe in the precision/recall curves that they achieve a sustained improvement to precision.

We observe that dHED-3D does not perform as well as either of the other 3D-CNN methods, providing evidence that directly merging lower resolutions into higher ones is not effective. Furthermore, precision-recall curves show that I2I-3D consistently improves precision across all recall values over other 3D-CNN frameworks. This indicates that our fine-to-fine multi-scale architecture increases localization resolution without loss of recall. In Figure 3.5, we see the qualitative results of our fine-to-fine architecture. We observe in this figure, particularly when compareing Figures 3.5(e) and 3.5(h), 3.5(f) and 3.5(i), that I2I-3D produces more precise boundaries than HED-3D.

3.6 Conclusion

We have introduced two network structures, HED-3D and I2I-3D, that address major issues in efficient volume-to-volume labeling. Our HED-3D framework demonstrates that processing volumetric data natively in 3D, has performance benefits over its 2D counterpart. Our I2I-3D framework efficiently learns multi-scale hierarchical features and generates precise

voxel-level predictions at input resolution. We demonstrate, through experimentation on two datasets, that our approach produces accurate and precise volume-to-volume labels. We compare our approach to a powerful 2D-CNN classifier, various strategies to multi-resolution merging with 3D-CNNs, as well as popular and state-of-the-art methods for each task. We provide our source code and pretrained models to ensure that our approach can continue to be applied to wide variety of medical applications and domains.

Acknowledgements

Chapter 3, in part, is a reprint of the material as it appears in Jameson Merkow, Alison Marsden, David Kriegman, and Zhuowen Tu. International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2016. The dissertation author was the primary author of this paper.

Chapter 3, in part, has been submitted for publication of the material as it may appear in Jameson Merkow, Alison Marsden, Zhuowen Tu, and David Kriegman. Medical Image Analysis, 2017 The dissertation author was the primary author of this paper.

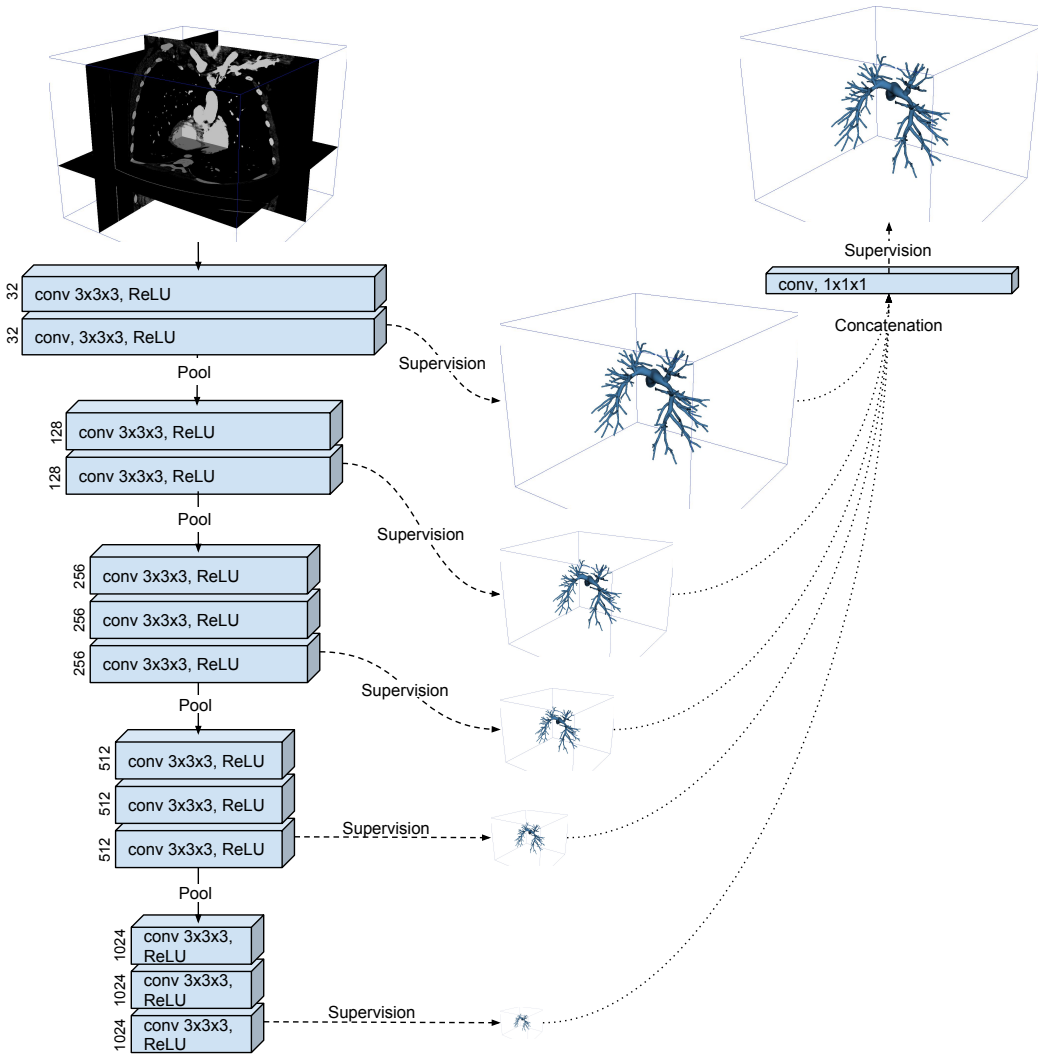


Figure 3.1. An illustration of the network architectures, HED-3D. HED-3D combines multiple side-outputs at different scales and uses simple aggregation to fuse them into a final output at the scale of the input volume. The number of channels is denoted on the left of each convolution layer and arrows denote network connections and operations.

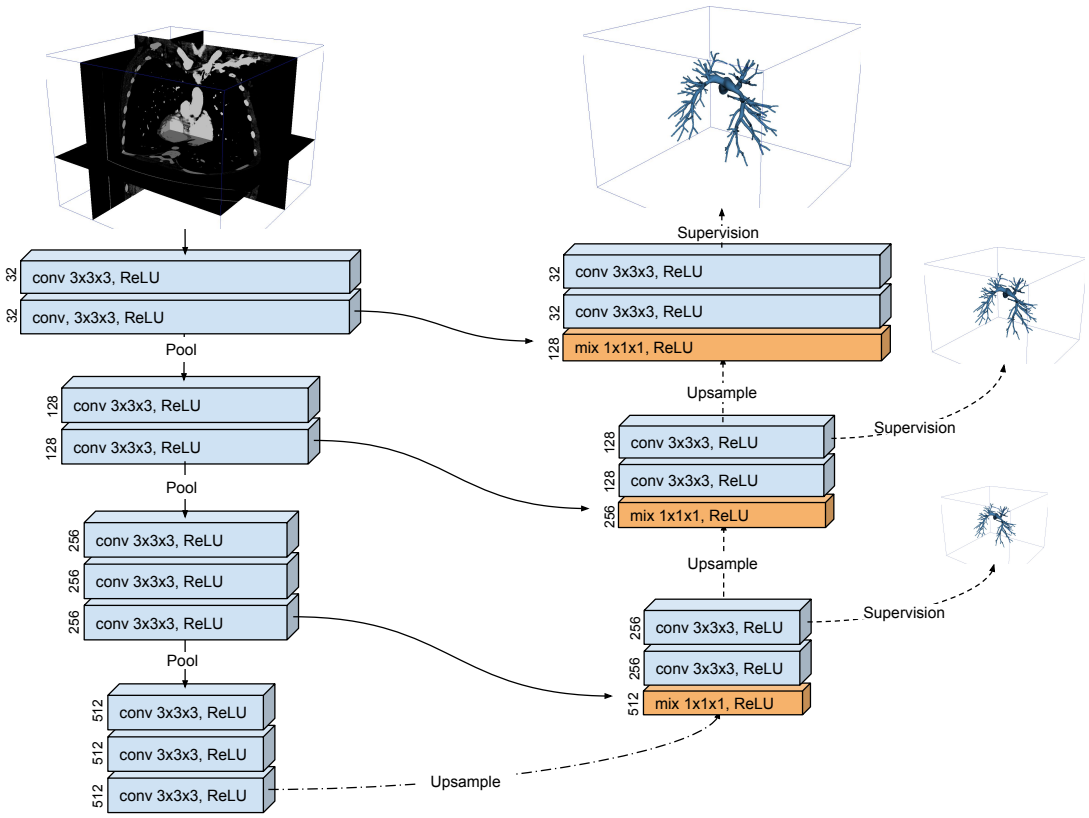


Figure 3.2. An illustration of our network architecture, I2I-3D. Our architecture couples fine-to-coarse and coarse-to-fine convolution structures and multi-scale loss to produce dense voxel-level labels at input resolution. The number of channels is denoted on the left of each convolution layer and arrows denote network connections and operations.

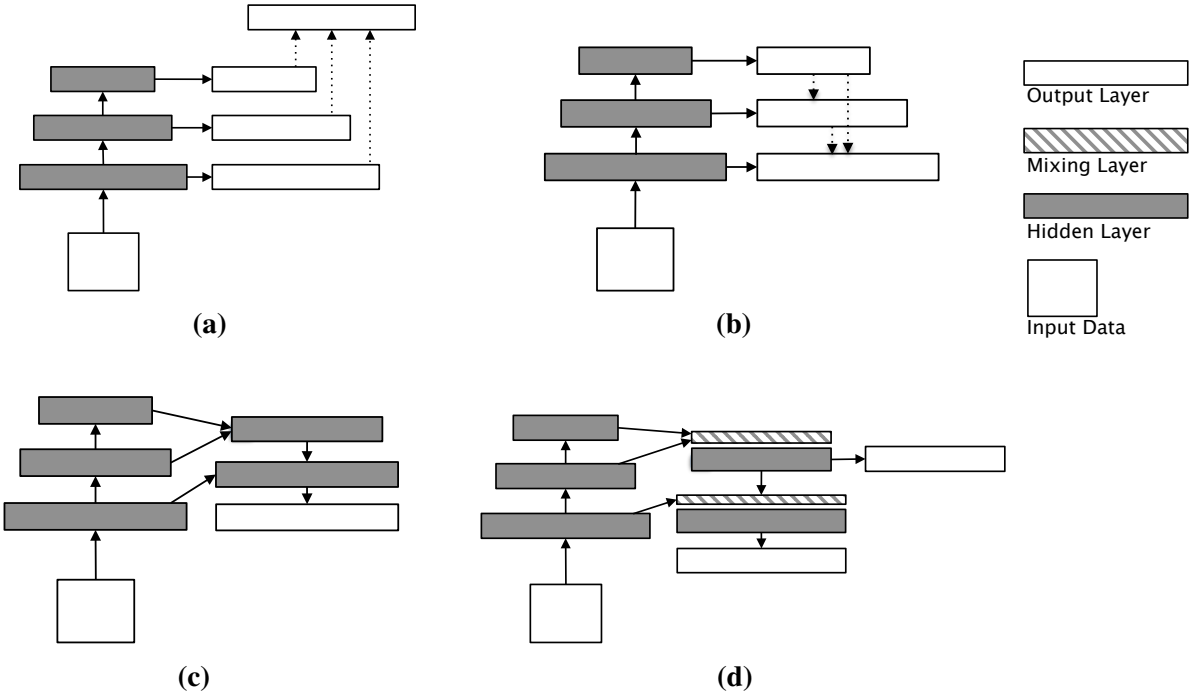


Figure 3.3. Depictions of different multi-resolution merge strategies. (a) downstream merging where higher resolution predictions are fused later in the network. (b) upstream merging where coarse predictions are connected directly to finer resolution predictions, earlier in the network, (c) feature merging where representations are merged rather than side-outputs. Lastly, the architecture in (d) combines multi-level features similar to (c), however, it also directly learns multi-scale features through mixing and applying multi-scale loss at each resolution. Method (a) corresponds to the strategy used by HED and HED-3D and is described in Section 3.3.1. As described in Section 3.3.2, method (b) illustrates the approach used by dHED and dHED-3D. Method (c) is the strategy used by UNet. I2I uses method (d), described in greater detail in Section 3.3.3.

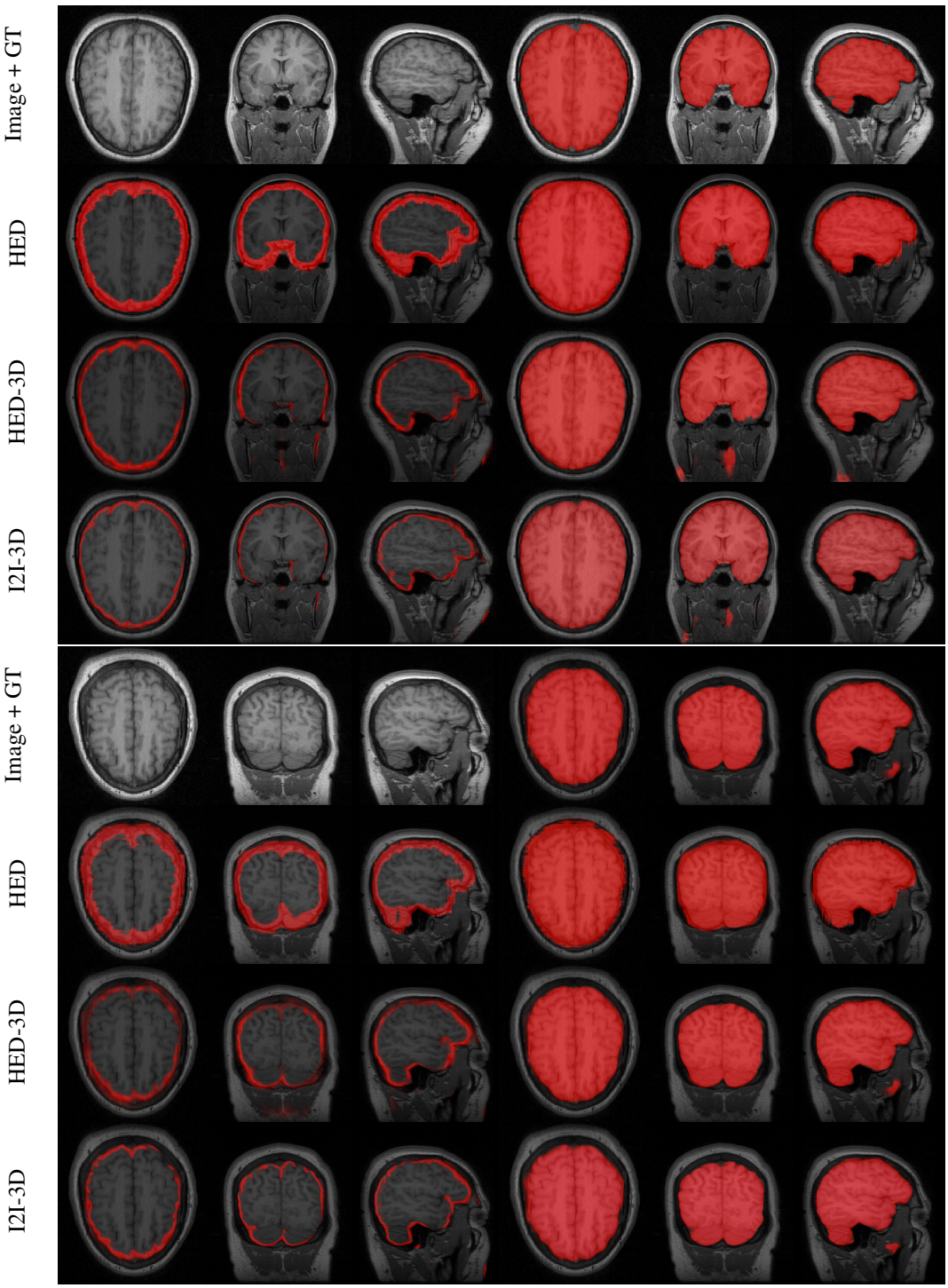


Figure 3.4. Illustration of boundary detection and skull stripping. The first row of each block shows the input volume and overlaid ground truth brain mask. In other rows, the first three columns depict boundary and brain mask predictions (overlaid) from each classifier.

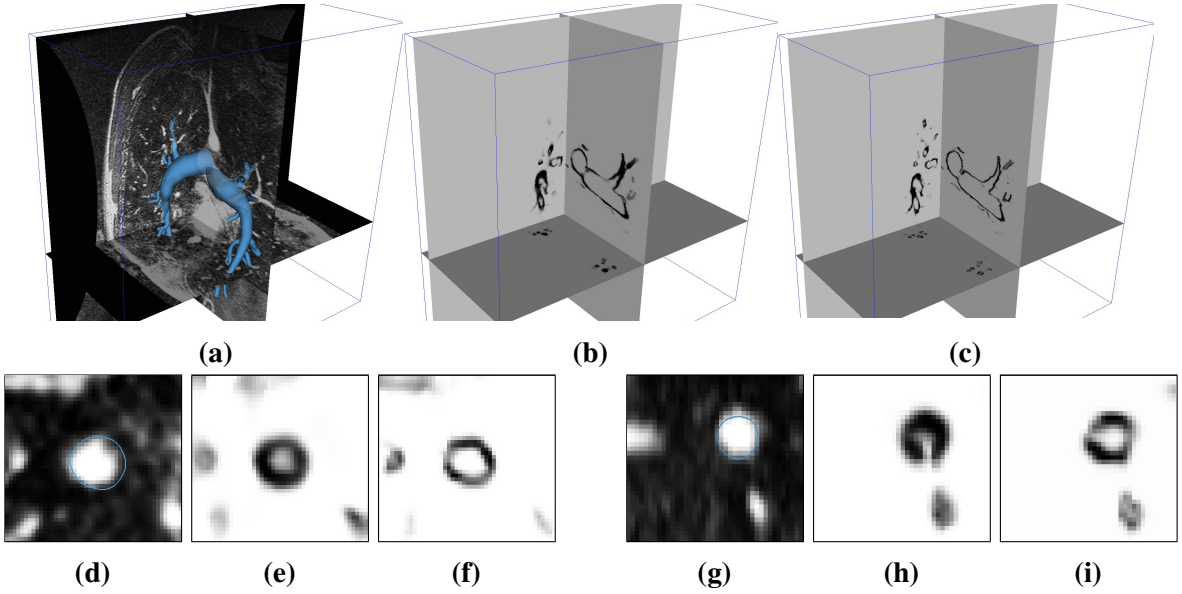


Figure 3.5. Results of our HED-3D and I2I-3D classifiers on vessel boundary detection. (a) Input volume and ground truth (in blue), (b) HED-3D result, (c) I2I-3D result. (d),(g) vessel cross section and ground truth (in blue). (e),(h) HED-3D cross section result. (f),(i) I2I-3D cross section result.

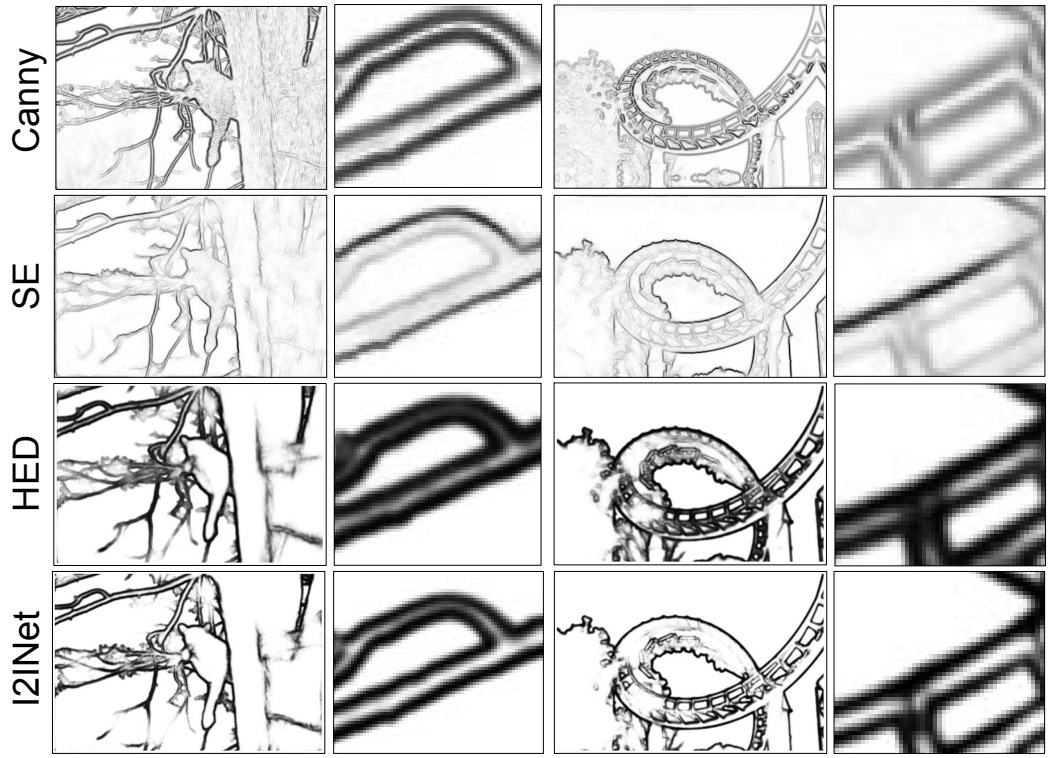


Figure 3.6. Illustration of the proposed I2I architecture. Each row depicts state of the art edge detection from different algorithms without non maximum suppress. We show clear advantage in accuracy over Canny and Structured Forests, and greater precision than HED, while maintaining overall performance.

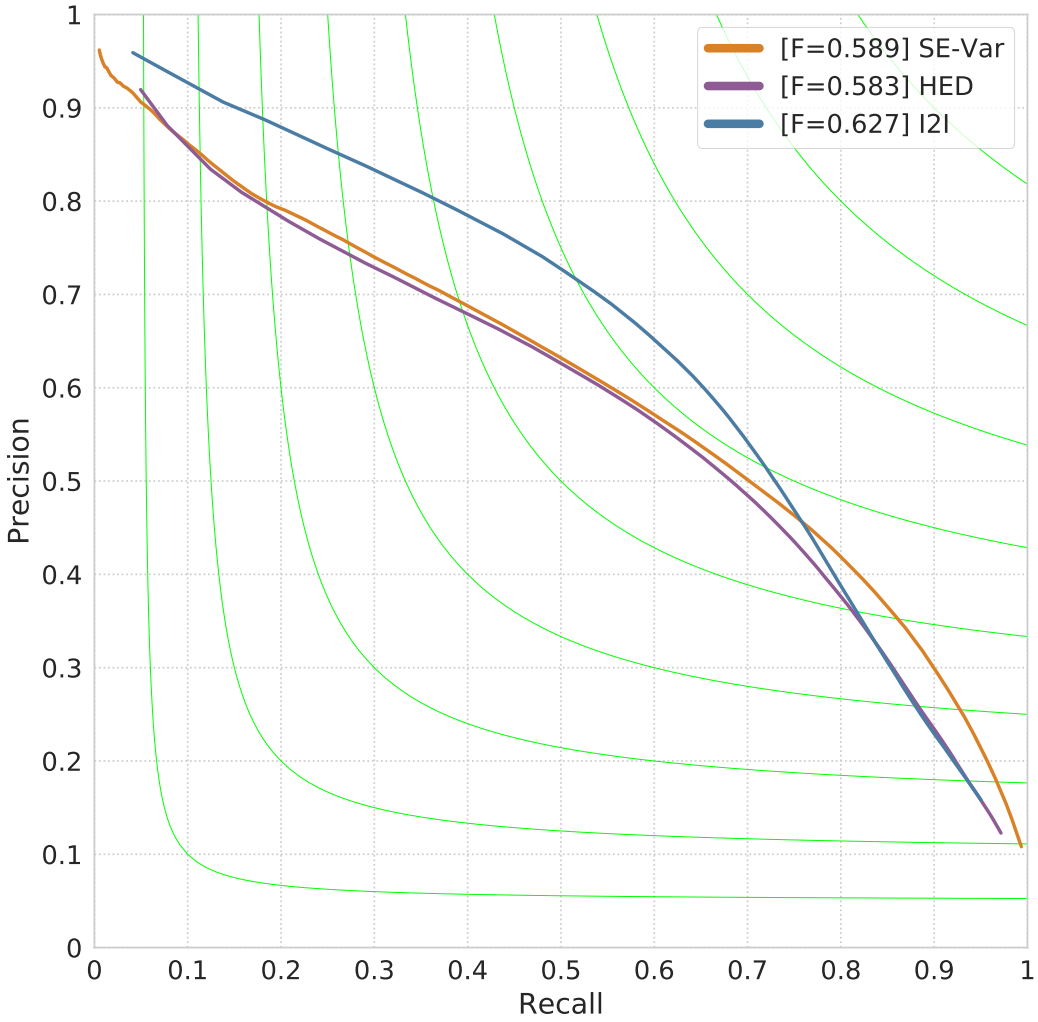


Figure 3.7. Precision recall curves comparing I2I-2D with the state of the art in natural image edge detection without non maximal suppression post-processing.

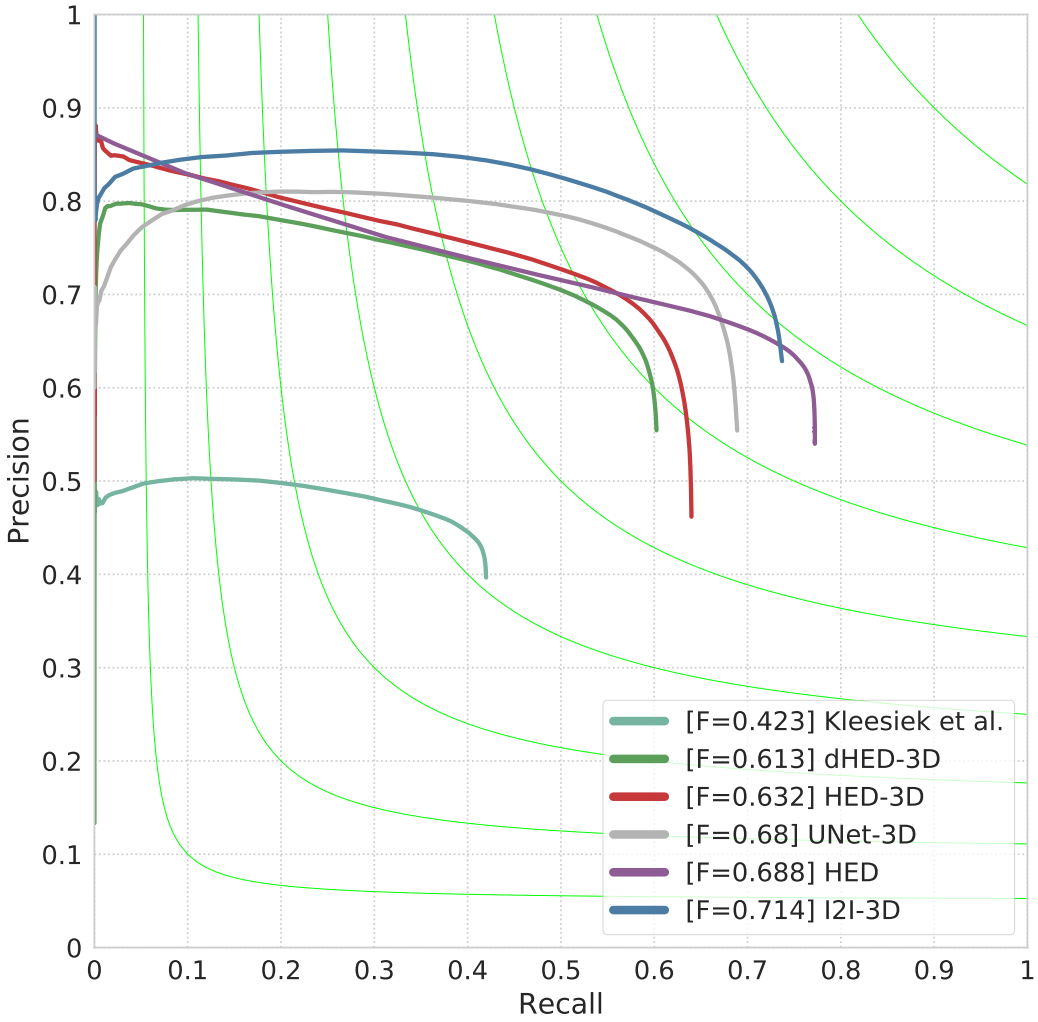


Figure 3.8. Results on brain boundary detection. Precision recall curves comparing our approach with baseline methods.

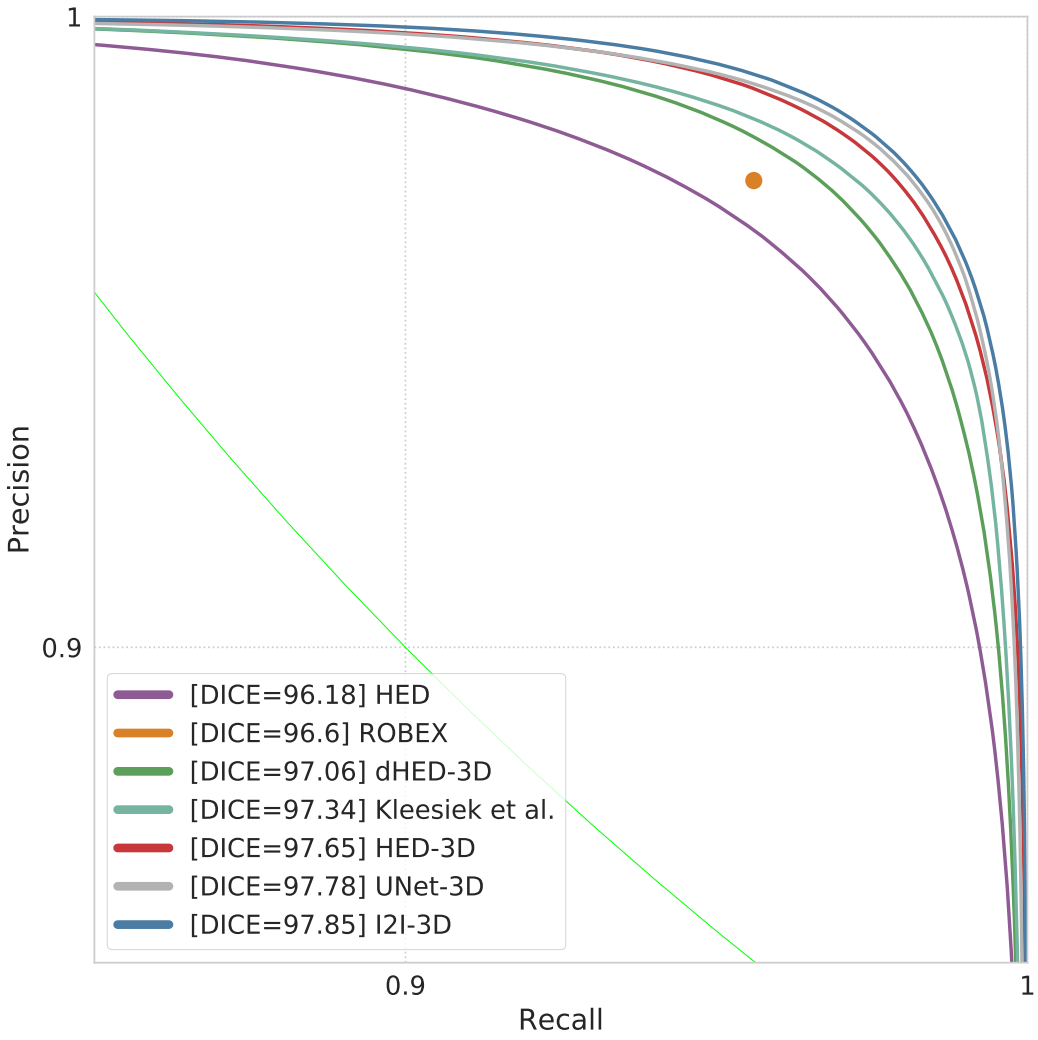


Figure 3.9. Results on skull stripping. Precision recall curves comparing our approach with popular methods and baseline methods.

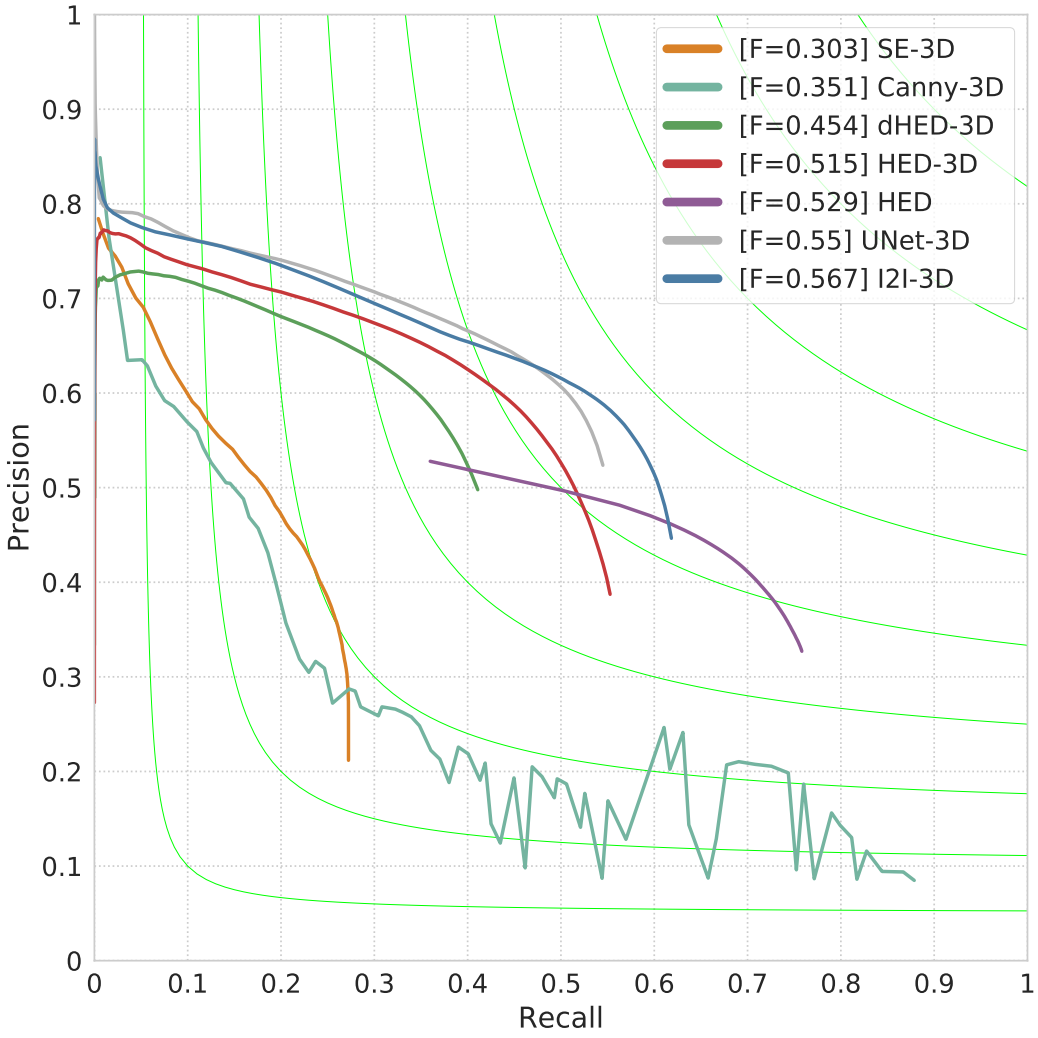


Figure 3.10. Results on vascular boundary detection. Precision recall curves comparing our approach (HED-3D and I2I-3D) with state-of-the-art and our baseline methods.

Chapter 4

Cardiovascular Model Construction with Convolutional Neural Networks

4.1 Introduction

Patient-specific simulations of cardiovascular hemodynamics [TF09] are gaining increased clinical utility for improving cardiovascular disease treatment [Mar13, TBT+14]. Hemodynamics simulations are currently used in myriad cardiovascular applications, including analysis of atherosclerotic plaque progression [SEM+11], development of novel surgical approaches for treatment of congenital heart disease [EMHMoCHAMI15], novel cardiovascular graft design [MBR+09], improved surgical planning [WTT+17] and as an accurate diagnostics tool for coronary heart disease [TFM13]. Simulation results have also been used to study intra-cranial and abdominal aneurysms [KSVS17, HT08], pulmonary flows and stenosis [dLDM+96, SKM+15], and coronary stents and grafts [MPC+02, GMY+12, RCSF13, RKM16]

To translate these tools to the clinic, studies on large patient cohorts are increasingly required in order to statistically correlate simulation outputs with clinical outcomes. However, production of accurate hemodynamics simulations requires construction of high quality three-dimensional patient-specific cardiovascular models. Cardiovascular models are typically built manually using a variety of image segmentation tools that generate vessel surfaces which

are refined and merged to form the final 3D cardiovascular anatomic model. Most available image segmentation algorithms require method-specific parameters, which require substantial tuning to produce accurate segmentations. This process is not only cumbersome and time consuming for non-expert users but also introduces variation into simulation results [NUZ00]. Manual model construction is similarly time-consuming and requires expert knowledge of cardiovascular imaging and anatomy. As a result, cardiovascular model construction currently represents a major bottleneck for large-scale studies requiring hemodynamic simulations to be performed in large cohorts.

Software packages such as SimVascular [UWM+13], the Vascular Modeling Toolkit (VMTK) [APB+08] and Cardiovascular Integrated Modeling and Simulation (CRIMSON) [KF16] specialize in cardiovascular model construction and blood-flow simulation. Typically, this process, (Figure 4.1), begins by loading medical image volume data and constructing pathlines along vessels, then building the model one vessel at a time through 2D segmentation of cross-sectional slices at discrete locations along the pathlines. Next, these 2D cross sections are oriented in 3D space and interpolated to form a vessel surface, and multiple vessels are merged, through Boolean operations, into a final 3D model.

In this work-flow, segmentation (Figure 4.1(c)) is by far the most time consuming step, often taking several days to build sufficiently complex models. In practice, users often opt to manually segment vessels due to inadequacy of current automated methods, which exacerbates this bottleneck, since a large number of segmentations (upwards of 50) are typically required per vessel and there are often 10-100 vessels per model. In contrast, manual pathline construction is a much simpler process, requiring identification of the vessels of interest with a few mouse clicks per pathline which requires minutes to a few hours.

Despite the availability of automated image segmentation methods, such as level sets, thresholding and region growing, they have not been widely adopted in the cardiovascular modeling community due to several drawbacks. First, these methods usually require more ef-

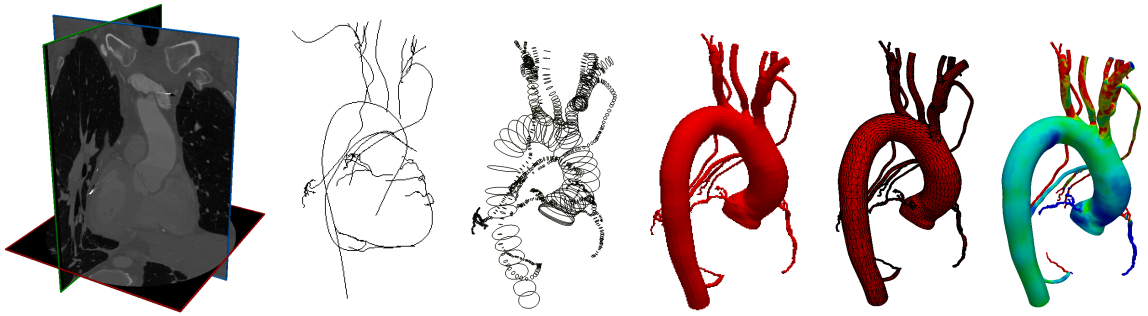


Figure 4.1. The cardiovascular model construction work-flow used in SimVascular [UWM+13]. Starting from (a) Image data, (b) users manually generate pathlines, (c) use these pathlines to segment 2D cross section contours, (d) loft segmented contours into a 3D model, (e) generate a numerically stable 3D geometric mesh, and finally (f) computational flow simulations are calculated.

fort and time to properly tune algorithmic parameters. Second, they often produce un-realistic segmentations that need to be manually edited. Recently, there has been an increase of machine learning approaches aiming to improve model construction efficiency through voxel and pixel level labeling [MMKT16, MTKM15, KLN+16, CALR17, SWR+16, BRLF13]. These methods use "ground-truth" segmented medical image volumes to train machine learning models or neural networks to perform medical image segmentation. These methods require no user interaction, a significant advantage that represents an opportunity to accelerate the cardiovascular model construction process. However, a fully end-to-end model construction methodology has yet to be proposed.

Our method, DeepLofting, combines a novel, spatially aware, CNN architecture with automatic contour generation to perform automatic model construction of 3D cardiovascular models from image and pathline data. By using our powerful CNN classifier, DeepLofting eliminates the need for parameter tuning and contour selection during vessel segmentation, allowing accurate and automatic model construction across a range anatomies and imaging modalities without human interaction thereby significantly accelerating 3D cardiovascular model construction. Central to DeepLofting is a novel neural network architecture, I2I-FC,

which utilizes pathline contextual cues to accurately produce precise localization of vascular structures. Our CNN embeds spatial processing into the I2I CNN classifier [MMKT16] to form I2I-FC. Our I2I-FC architecture refines the prediction by the base I2I classifier by enhancing vessel detection and ignoring extraneous vessels. From these segmentations, contours are autonomously extracted and oriented into 3D space to create vessel surfaces. Multiple vessel surfaces are automatically merged through Boolean operations to form a final 3D model. Our method requires substantially less user involvement and represents a critical step forward in automatic 3D model generation.

4.2 Background and Related Work

4.2.1 Cardiovascular Model Construction

Construction of cardiovascular models from medical image data is a complex multi-step process. Image segmentation plays an important role in all cardiovascular modeling algorithms, however segmentation strategies differ depending on the application. These strategies can be separated into two broad categories, 1) direct 3D methods, which generate a model directly from image data and 2) 2D path-planning methods, which use the pathlines of vessels to guide model construction. Here, we review relevant work and refer the reader to comprehensive reviews of the various stages of cardiovascular model building for additional information [SLLL02, LABFL09, Duf13].

Direct 3D Cardiovascular Model Building

In cardiovascular imaging, contrast enhancing agents are combined with magnetic resonance (MR) or computed tomography (CT) imaging to produce 3D medical image volumes where blood vessels produce high pixel intensities. Direct 3D cardiovascular modeling methods leverage this property to construct approximate 3D surfaces of vessels for a given image.

Various strategies have been used to develop 3D cardiovascular model building methods. For example, [WLK+11] use a level set based algorithm that numerically solves partial differential equations and propagates an initial surface to regions of high contrast. Other methods use filters to enhance the pixel response of vessels allowing segmentations to be obtained through thresholding. For example [FNVV98] uses the local pixel intensity Hessian and [Law08] use the flux of image intensity along specific vectors to increase vessel visibility. Image enhancement methods are often combined with active contours such as level sets to improve performance for detection of the structures of interest [LC10, SDN+11].

Another common approach represents voxels as a graph of connected "nodes" and edge and node values are used to partition a volume. Graph based methods, such as normalized cut [Shi00], have been extended to vascular segmentation by incorporating information such as tubular shape priors [Bau09], vessel-ness filter response [Wan16] and Hessian tensors [WKN+16]. Other graph-based methods use a curvilinear computation and integer programming [Tur13, Rob16] to identify tubular structures for vessel segmentation. Whereas other approaches use a max-flow based optimization scheme which, when combined with Hessian vessel-ness filters, has also been used for tubular structure segmentation [Pez16].

Direct 3D model building methods avoid the need for manual model construction, making them the method of choice for most commercial and open-source medical image processing software such as 3D Slicer [FBKC+12], the Insight Registration and Segmentation Toolkit (ITK) [YAL+02], ITK-Snap [YPCH+06] and the Vascular Modeling Toolkit (VMTK) [APB+08]. However direct 3D methods often require substantial image pre-processing and tuning of method specific parameters on a case-by-case basis. This currently makes high throughput cardiovascular model construction difficult to achieve with direct 3D methods.

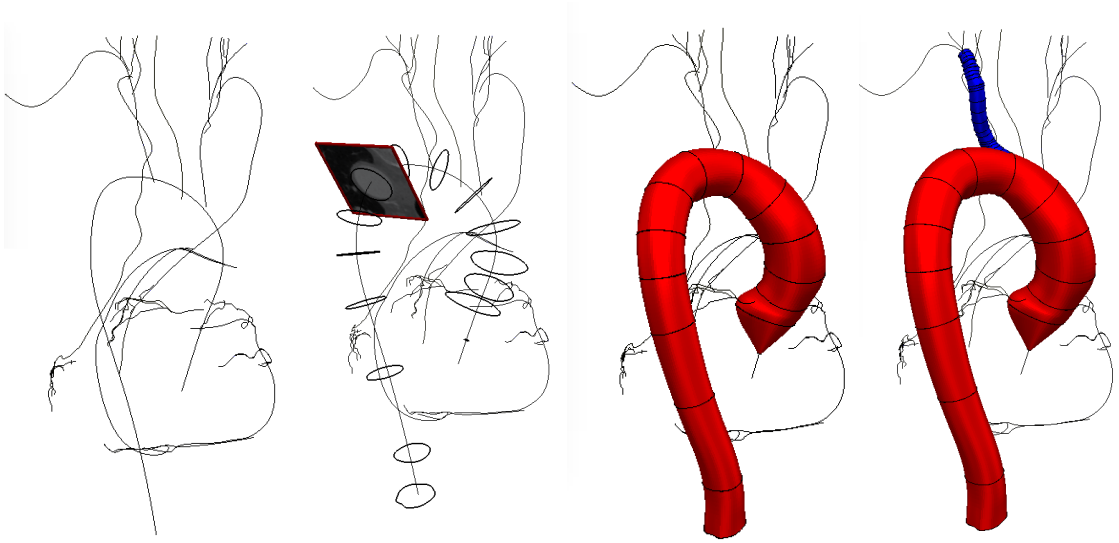
Path-planning Model Building Methods

Path-planning uses a different approach and treats blood vessel networks as individual tubular structures which are segmented by navigating along their path and merged to form a final anatomic model. Pathlines indicate the path followed by particular vessels in the medical image volume and traversing a volume along these pathlines results in a cross-sectional view where one can easily view and outline the vessel lumen. Using pathlines to segment the each vessel constrains the model construction process thereby reducing difficulties in accurately localizing vessel surfaces.

Figure 4.2 shows a typical pathline based model building procedure used by multiple software packages [UWM+13, KF16]. This work-flow follows four major steps: (1) pathline annotation, (2) vessel cross-section segmentation, (3) 3D lofting and interpolation, (4) vessel union. As with 3D model building, various strategies exist to improve efficiency for these stages.

Path-planning methods for model construction start with annotated vessel pathlines, typically performed by a human user [PTW98, TF09]. The user selects the approximate pathline of each vessel by picking control points along the vessels; from these a smooth line is interpolated. Typically, pathlines produced by pathline extraction algorithms are not sufficiently accurate for model construction, and require substantial manual corrections. Therefore users typically manually specify the pathline as it only requires selection 10-20 control points. These control points are interpolated, using splines to 100-300 points which form the basis of cross-section segmentation along the vessel path.

Next, segmentations are generated at selected points along the interpolated pathline from cross-sectional 2D images extracted from the plane perpendicular to the pathline direction. Vessel cross sections are segmented using manual annotation or two dimensional segmentation algorithms such as level sets or thresholding [Wan01, LXGF10]. Manual contour labeling requires selection of 15-25 contour points and a typical vessel requires 30-50



(a) Path creation. **(b)** Cross section segmentation. **(c)** Lofting segmentations. **(d)** Union of lofted surfaces.

Figure 4.2. Illustration of model segmentation process. Creation of a vascular geometry using the lofted 2D segmentation approach involves (a) creating path to navigate and create a series of segmentations (b) that are lofted to form each vessel (c). A solid model is generated by the union of individual vessel models (d). This figure is reproduced from [UWM+13] with author permission.

cross-section segmentations to build an accurate 3D vessel surface. Automated segmentation algorithms may be well suited for larger vessels or vessels with high intensity contrast, however, other vessels typically require manual segmentation, as these segmentation algorithms often produce unacceptable errors which necessitate substantial manual correction.

The next step, "lofting", 3D surfaces of each vessel are produced by orienting the vessel cross section contours along the pathline (Figure 4.2(b)) and interpolating a surface along the associated pathline (Figure 4.2(c)). After all vessels of interest are modeled, they are merged through Boolean operations as shown in Figure 4.2(d). Once the final cardiovascular model is generated, it is converted into a numerical mesh, boundary conditions are assigned to inlets and outlets of the model and a computational fluid dynamics (CFD) solver carries out a blood-flow simulation [UWM+13].

There have been many approaches for reducing the human burden in the time-

consuming segmentation step. For example, model-based approaches define vessels as curves with circular or elliptical cross-sections with radii that vary along the curve [Kri00, LY07, MST10, BC11]. Vessel models are fit to the image data by minimizing an objective function corresponding to the structure of interest. Other methods construct vessel templates and identify where along the pathlines to place these templates. For example, [ZBG+07] extracts a vascular skeleton from a 3D segmentation to construct a model by sweeping non-uniform rational B-splines (NURBS) vessel templates along that extracted skeleton. Yet another approach applies this sweep template strategy to an implicit function to build cardiovascular models along the pathline [KGPS13]. In [KYD+17], a pathline tracking algorithm, initialized from user supplied seed points, extracts a vessel skeleton, after which local implicit functions model the vessel surface.

Despite constraining the model via the pathlines, many pathline-based methods suffer from drawbacks similar to 3D model building methods. 2D segmentation methods still contain method-specific parameters which must be repeatedly tuned to obtain accurate segmentations across images, volumes and anatomical regions. Thus, existing pathline based methods have failed to produce a high throughput cardiovascular model construction strategy.

4.2.2 CNN Segmentation

Recently, learning methods have come to the forefront of natural image and medical segmentation strategies. Learning based methods learn directly from image data and, once trained, require no parameter tuning, in contrast to other segmentation methods. In addition, learning based methods can be modified to utilize multiple sources of information. For example, in a recent decision tree based approach to vessel wall localization, [MTKM15] used image data alongside pathline based atlas features to robustly detect vessel boundaries in 3D. However, classical machine learning algorithms which require significant feature engineering have fallen to the way-side in favor of CNN-based segmentation algorithms

capable of learning from raw input data.

Fully convolutional networks (FCN) [LSD15] demonstrated efficient holistic image segmentation with CNNs by replacing fully connected layers with convolutional layers in the VGGNet architecture [CSVZ14]. Spurred by the success of FCN, Holistically Nested Edge Detection (HED) [XT15] used multi-scale fusion and deep supervision to obtain improved segmentation accuracy on fine-scale image structures to obtain near human-level performance in edge-detection. U-Net [RFB15] extended FCN to medical imaging by adding concatenation and convolution layers in the expanding path of the network so finer-scale details could be captured. More recently, medical imaging research efforts have focused on extending FCN architectures to 3D volume segmentation. In [KLN+16], multi-resolution FCNs were used for brain tumor segmentation from 3D medical image volumes. A 3D U-Net architecture was developed and applied to kidney segmentation, making additional use of data augmentation and developing techniques to learn from sparsely labeled planar views of 3D medical images [CALR17]. The I2I architecture [MMKT16] achieved state-of-the-art performance in volume-to-volume vascular boundary detection, by enhancing segmentation precision through coupled efficient coarse-to-fine and fine-to-coarse paths that targeted multi-resolution learning.

CNN methods have shown their effectiveness for image and volume segmentation, however, these methods have yet to be applied directly to cardiovascular model construction. In addition, CNNs in the FCN family sacrifice image-wide spatial context for efficiency by replacing fully connected layers with convolution layers. This approach is suitable for tasks where the segmentation can appear anywhere in the image, however this context is critical in applications where location is discriminative such as cardiovascular model construction.

4.3 Methodology

Although convolutional networks have been widely used for image segmentation, little work has employed neural networks for anatomical model construction. In this section,

we describe our neural network architecture and its application within a framework for cardiovascular model construction.

4.3.1 Problem formulation

We begin by providing a mathematical formulation of our approach. Our goal is to segment a full 3D model $\mathbf{Y}$ from a given volume, V, and a set of pathlines, $\mathscr{P}$. This goal is broken down into sub-tasks where each vessel Y_k is segmented individually, then merged to form the final model $\mathbf{Y}$. Specifically, we are given an image volume, $V \in \mathbb{Z}^{W \times H \times D}$, and a set of K corresponding path-lines, $\mathscr{P} := \{P_1, ..., P_K\}$ where P_k represents the kth path-line and P_k is a sequence of coordinates, $P_k := \{p_{k,j}\}, j = 1, ..., M$, where $p_{k,j} \in \mathbb{R}^3$.

For each pathline, P_k, there is a ground-truth vessel segmentation, $Y_k \in \{0,1\}^{W \times H \times D}$, where $Y_k(x,y,z) = 1$ when voxel $V(x,y,z)$ is inside the vessel corresponding to pathline k and $Y_k(x,y,z) = 0$ otherwise. In other words, Y_k is a binary segmentation of only the kth vessel.

We seek to estimate $\mathbf{Y}$ from the input volume V and set of paths $\mathscr{P}$ by merging a corresponding set of predicted vessel segmentations, $\hat{\mathscr{Y}} := \{\hat{Y}_1, ..., \hat{Y}_k\}$ such that each $\hat{Y}_i$ is an accurate estimation of ground truth Y_i. The final model is produced by $\hat{\mathbf{Y}} = \text{UNION}(\hat{\mathscr{Y}})$ where $\text{UNION}(\cdot)$ represents a surface Boolean operation that joins vessel surfaces. Our approach reduces the estimation of each $\hat{Y}_k$ to cross sectional segmentations along the pathline P_k for which we use a convolutional neural network specialized to incorporate image-wide spatial context into its predictions.

4.3.2 Spatially Aware CNNs for Segmentation

Our methodology relies on precise and accurate segmentation of individual vessels at 2D cross-section images along the path. While fully-convolutional architectures achieve excellent performance for pixel level labeling, they are unable to capture spatial positioning information since convolution is a spatially invariant operation. Fully convolutional networks

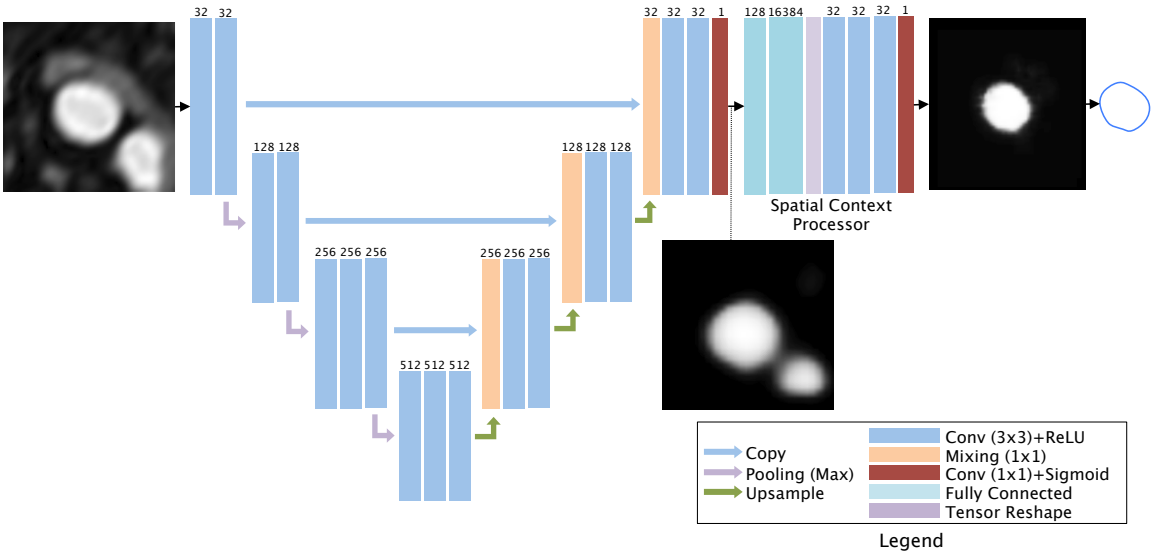


Figure 4.3. An illustration of our spatial context network, I2I-FC. The spatially invariant FCN segmentation is followed by our proposed spatial context processor which computes a localized encoding using fully-connected layers. The final output is an enhanced localized segmentation.

sub-optimally capture contextual cues based on image structure such as those in vessel cross-section images sliced along pathlines. In particular a previous approach trained a 3D fully convolutional multi-resolution network, named I2I-3D, to detect pixels corresponding to vessel boundaries in 3D medical image volumes [MMKT16]. However, despite improving vessel edge detection accuracy significantly, predicted edge maps were often still multiple pixels wide and not straightforward to convert into accurate cardiovascular models.

We address the short comings of previous attempts at using CNNs for model construction in two key ways. First, we train CNNs to produce vessel segmentations as opposed to pixel-wise edge predictions, simpifiing the extraction processing by using a simple marching-squares algorithm. Second, our proposed spatially aware classifier is designed to utilize image-wide spatial context leading to enhanced vessel localization. For our base classifier we used I2I [MMKT16], which allows our method to benefit from the increased accuracy and precision provided by I2I's multi-resolution architecture.

In our novel CNN architecture, images pass the I2I network to produce an initial

segmentation prediction which is refined by two fully-connected layers and three convolution layers and final 1×1 convolutional layer. The first fully connected layer transforms the prediction of I2I from $128 \times 128 = 16384$ into a smaller 128×1 dimensional tensor. Next, second fully connected layer transforms this tensor back into 16384 elements which is reshaped back into a 128×128 tensor. Since fully connected layers are computationally expensive, we use only two fully connected layers to encode spatial context. The fully-connected layers ensure that each output segmentation uses information from the entire FCN segmentation and allows the network to more precisely localize structures of interest. Since the spatial context processor is a part of the network it can be trained end-to-end allowing the network to learn to process spatial context from the data and no post-processing is required.

4.3.3 The DeepLofting Pipeline

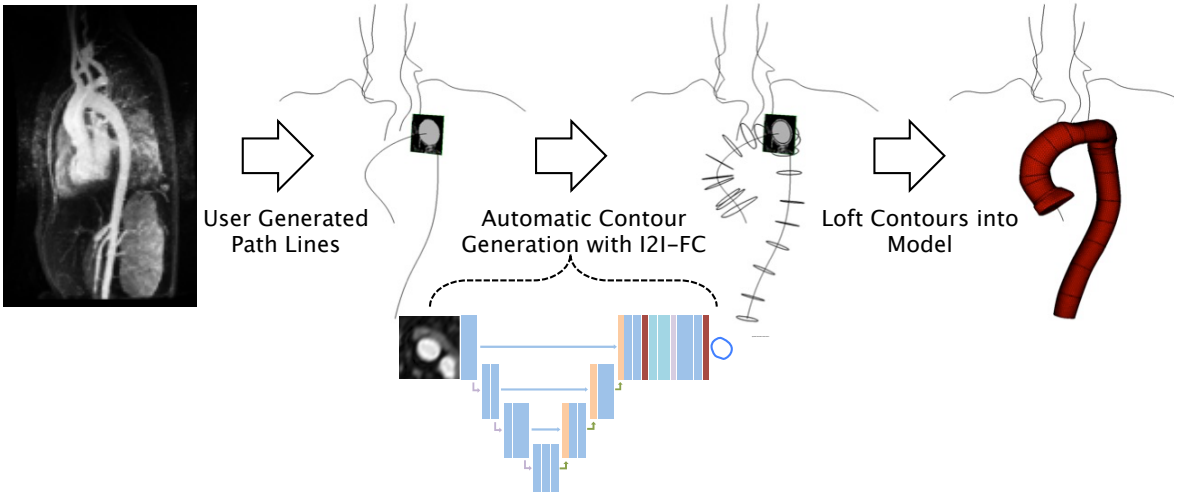


Figure 4.4. An illustration of our DeepLofting pipeline. Path-lines are built from a medical image volume, thereafter image patches are extracted along the path-lines and automatically processed by our CNN and lofted into a 3D model. The resulting segmentations are converted to vessel boundaries and reoriented along the path-lines. In the final step, the vessel boundaries are lofted together to form a solid model.

Our spatially-aware CNN architecture computes precisely localized segmentations of 2D images and incorporates spatial context such as vessel pathline locations as part of our

automated 3D cardiovascular model construction pipeline, DeepLofting shown in Fig. 4.4. DeepLofting begins with creation of vessel pathlines to specify which vessels/region are of interest. Typically, pathlines are built by selecting a modest number of control points within a medical image volume and generating a full path through interpolation. These pathlines provide context within a medical volume, specifying which vessels to include in the full 3D cardiovascular model. DeepLofting uses the pathlines for vessel selection and derives spatial cues to improve the standard lofting work-flow (depicted in Figure 4.1) by eliminating the need for user-intervention in segmentation.

Image patches are extracted at points centered at points along the interpolated path which are then automatically segmented with our spatially aware CNN, I2I-FC. Since I2I-FC is trained to produce a response for only a single vessel, loft-able contours can easily be extracted from the segmentations using marching-squares. Once contours are generated, they are lofted into 3D space along the path and a 3D surface is interpolated. The process is repeated for each pathline. Lastly, we use specialized Boolean operations [UWS16] to union all vessel surfaces into a complete cardiovascular model.

4.3.4 DeepLofting Training Procedure

Our input training set consists of N 3D volumes $V := \{v_1, ..., v_N\}$ where $v_i \in \mathbb{Z}^{W_i \times H_i \times D_i}$. Associated with each image is a set of K path-lines $\{P_1, ..., P_K\}$ where the kth path-line represents a sequence of (x, y, z) physical space coordinates in the image $P_k := \{p_{k,j}\}, j = 1, ..., M$, and $p_{k,j} \in \mathbb{R}^3$. Image and segmentation pairs, $x_{n,k,j}$ and $y_{n,k,j}$, represent 2D images captured at point $p_{k,j}$ in volume V_n. Each image pair is processed independently once it is extracted, and we denote these pairs with a simplified subscript: x_i and y_i, where i denotes the unique (n, k, j) triplet. Each image and segmentation pair, x_i and y_i have specified width w and height h. Pixel values are interpolated from the lumen value in the plane orthogonal to the path line direction. Spacing and pixel values are normalized. All segmentations are binary

representations of user annotated contours at the specified path point such that $y_k \in \{0,1\}^{w \times h}$ is a binary image where a pixel value of 1 denotes vessel and 0 denotes non-vessel tissue. At each path-point a contour is extracted from the 3D vessel surface and then converted into a binary image. We obtain our final dataset by grouping the image and segmentation pairs into a collection denoted by $X := \{(x_1, y_1), ..., (x_{|X|}, y_{|X|})\}$. This collection is used to train a Neural Network classifier, $\text{NET}(x, w) : x \to \{0,1\}^{w \times h}$, with m layers. The weights, w, are found by approximately minimizing a binary cross-entropy loss function on the dataset X. We calculate pixel-wise loss across a mini-batch with cross-entropy cost function:

$$\mathbb{E}_{x,y \sim X}[L(x,y,w)] \approx \frac{1}{N_{batch} hw} \sum_{i=1}^{N_{batch}} L(x_i, y_i, w),\ x_i, y_i \sim X, \tag{4.1}$$

where $L(x_i, y_i, w)$ is given by

$$-\sum_{j=1}^{w} \sum_{k=1}^{h} y_{ijk} \log\left(\Pr(y_{ijk} = 1 | x_i, w)\right), \tag{4.2}$$

, where y_{ijk} denotes the value of pixel (j,k) of y_i. $\Pr(y_{ijk} = 1 | x_i, w) = \sigma(a_{jk}) \in [0,1]$ are classifier predictions and a_{jk} is the pixel value at (j,k) of network output, a.

During training, parameters are updated using an Adam optimization [KB15] to minimize pixel-wise loss in (4.1). Given an image x_i, the estimated segmentation is denoted $\hat{y} = \text{NET}(x, w)$. Our networks are trained and evaluated on typical path-lines generated from multiple users which have varying degrees of accuracy. This makes the trained network inherently robust to image acquisition noise and inaccuracies in the user-specificed path-lines.

4.3.5 Cardiovascular Model Construction with DeepLofting

Using a trained classifier, the DeepLofting pipeline takes an input volume V and a collection of path-lines P to produce a final 3D model. We obtain a sequence of spatial coordinates, $\{p_1 ... p_{N_P}\}$, by selecting points at intervals along each path-line from P. At each

point p_i, we extract a corresponding image patch, x_i, by interpolating voxel intensity values of V in the plane normal to the path direction at p_i. For each x_i, we compute an estimated segmentation $\hat{y}_i$ by feeding x_i through our trained network. We then apply marching-squares to each $\hat{y}_i$ (using a fixed isovalue) to produce contours c_i. Appropriate isovalues are obtained through validation and are fixed for each net. The final solid model is constructed placing each contour, c_i, at its associated point p_i (in 3D), which orients c_i in the same plane as the original extracted image, x_i. The set of $C = \{c_1, ..., c_{N_P}\}$, are interpolated along the associated path, P, to form a complete vessel surface. Once all vessel surfaces are constructed, they are merged through a union operation, creating a complete 3D model.

The pathline based approach of DeepLofting has multiple beneficial properties. Our spatially aware CNN classifier produces consistent contours along each vessels resulting in a smoother final surface. DeepLofting is more efficient than volume-to-volume methods as only sparsely placed 2D images need to be segmented and therefore it avoids the computationally demanding task of segmenting entire volumes. In an application setting, users have fine-tuned control over what contours to generate, reducing this cost even further. In addition, our 2D CNN uses substantially fewer parameters, thus requiring less memory than an equivalent 3D CNNs allowing them to be used on consumer grade hardware.

4.4 Data

We evaluate our method on a dataset consisting of 100 contrast-enhanced medical volumes which contains an even split between CT and MR data. Each of the 100 volumes has a corresponding cardiovascular model and a set of vessel pathlines. Figure 4.5 shows the distribution of cases by anatomical region. All volumes in our dataset are publicly available from the Vascular Model Repository [1] [WOJ13]. This dataset contains image data, models and hemodynamics simulation results for a range of cardio-pulmonary regions, including

[1] http://www.vascularmodel.com

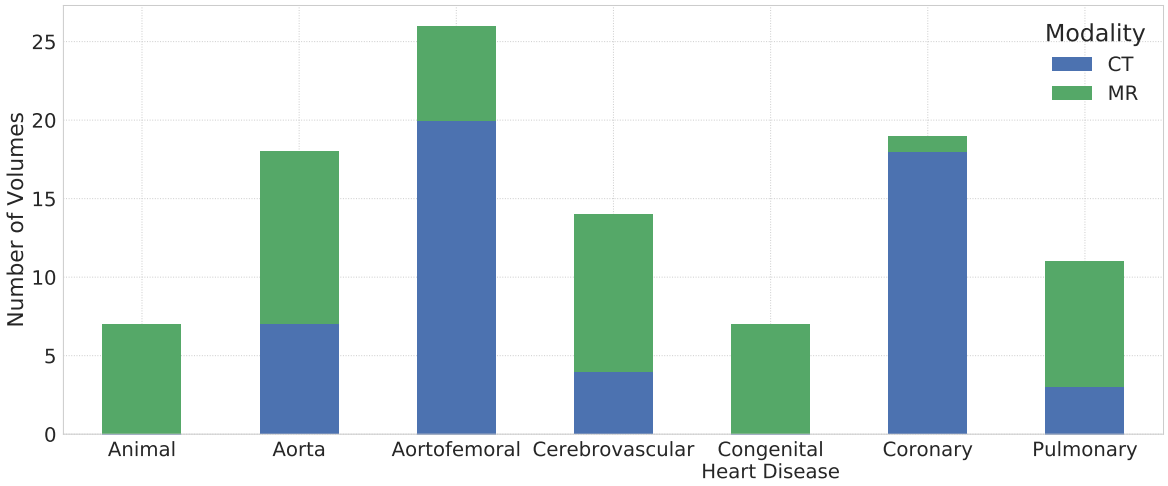


Figure 4.5. Distribution of anatomical region of included in our dataset of 100 medical volumes (50 MR and 50 CT). Volumes may span multiple regions. All volumes are available at the Vascular Model Repository [WOJ13]

both normal and abnormal physiologies. All segmentations were created in SimVascular using a pathline based model building approach by expert curators and verified by clinical collaborators. Approximately 25% of the cases have normal anatomy the remaining contain pathological models including abdominal aortic aneurysms, aortic coarctation, coronary aneurysms, coronary artery bypass graft surgery patients, and cerebral aneurysms [WOJ13]. Most image volumes in the Vascular Model Repository have anisotropic pixel resolution which we normalize during pre-processing. For experimentation, we split volumes into three mutually exclusive sets for training, validation, and testing of 76, 8 and 16 volumes respectively. Since the number and length of the pathlines vary model to model, the total number of cross section slices also vary slightly. After extracting image cross sections along pathlines, these volumes resulted in $108,253$ (CT) and $143,394$ (MR) images for training, $6,835$ (CT) and $12,812$ (MR) images for validation and $42,694$ (CT) and $53,987$ (MR) images for testing.

4.4.1 Data Pre-processing

During data pre-processing, we normalize the image intensity of all volumes before extracting 2D image patches along pathlines. For each modality, we apply different intensity normalization strategies.

In CT image volumes, pixel intensity corresponds to Hounsfield units, so as to not degrade this anatomical information, we scale all CT images using min-max normalization:

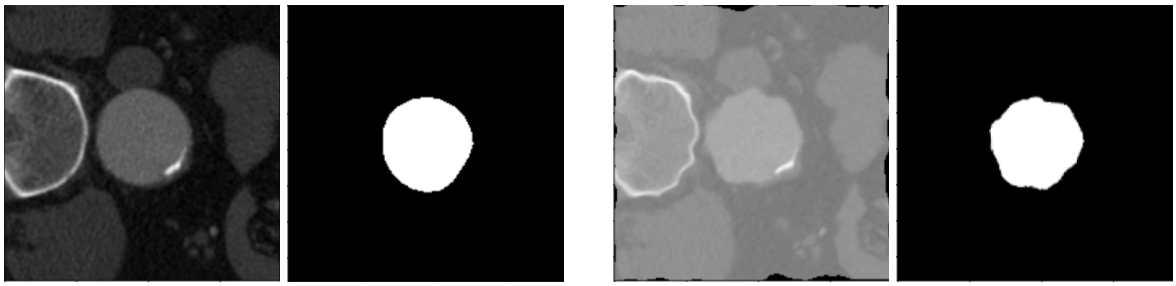
$$X_{CT,\text{norm}} = \frac{X_{CT}}{I_{CT}} \tag{4.3}$$

where X_{CT} is an unnormalized CT image volume, $X_{CT,\text{norm}}$ is a normalized CT image volume, and I_{CT} is a predefined intensity scaling value used for all CT image volumes. Given that Hounsfield units typically have a range between $\pm 3000HU$, we use $I_{CT} = 3000$ so that normalization would result in a range of ± 1.

Unlike CT image volumes, the pixel intensities in MR images do not have a fixed range, this makes defining a suitable normalization scheme for MR images difficult [NUZ00]. As such, for MR images we use per-image min-max normalization, where the minimum and maximum intensity scaling values are taken from each volume individually.

4.4.2 Data Augmentation

We use extensive data augmentation on each batch of images during training all neural networks. Specifically, image patches and corresponding ground truth segmentations are randomly rotated, shifted and transformed using elastic deformations [SSP03]. An example of augmentation is depicted in Figure 4.6. Augmentation is computed on-the-fly, allowing the training dataset to be expanded indefinitely.



(a) (b)

Figure 4.6. Example augmentation. (a) Regular image patch and ground truth segmentation from a CT image volume. (b) Image patch and ground truth segmentation after applying elastic deformation.

4.5 Experimentation

4.5.1 Evaluation Methodology

To evaluate our methodology, we measure performance of the segmentation process as well as the final 3D model produced by the DeepLofting pipeline. First, the 2D segmentation results are evaluated using three metrics: 1) the dice-coefficient (DICE), 2) Hausdorff distance and 3) Average Symmetric Surface distance (ASSD). Precision recall curves are calculated via a threshold sweep of all classifier predictions which are compared on binarized ground truth images. Second, we measure performance of the entire DeepLofting pipeline using each segmentation technique. The same performances metrics are calculated, and as with 2D evaluation, ground truth and predicted 3D models are converted to binary volumes before comparison.

Performance Metrics

Here, we briefly discuss our performance metrics which are illustrated in Figure 4.7. DICE is a commonly used metric for evaluating segmentation performance, however, it does not capture some key aspects for measuring 3D cardiovascular model fidelity. For these reasons, we also measure performance using Hausdorff distance, and ASSD. Hausdorff

distance is useful to measure the worst-case distance error between two binary segmentations. ASSD measures the average surface distance between two binary segmentations. Both ASSD and the Hausdorf distance are useful for evaluation as they take the physical image spacing into account.

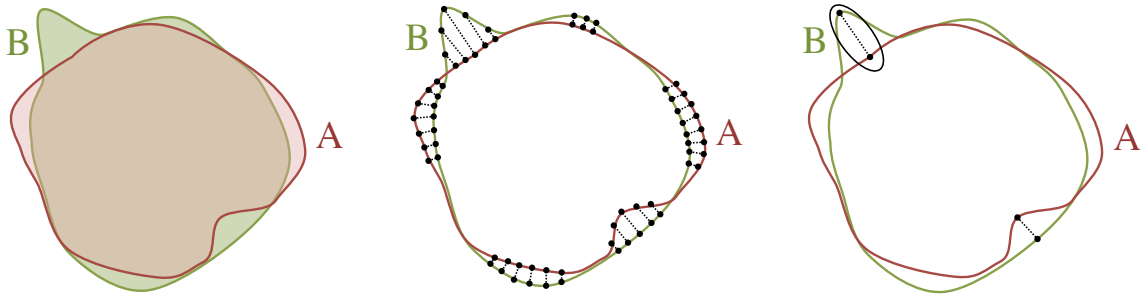


Figure 4.7. Illustrations of DICE coefficient, ASSD and Hausdorff distance metrics. (a) The DICE coefficient measures the ratio between the area of the intersection and union of two volumes. (b) ASSD measures the average distance between two surfaces. (c) The Hausdorff distance measures the maximum of the smallest distances between two surfaces.

DICE The DICE coefficient values range from 0 (no overlap) to 1 (no error) which measures the similarity between two segmentations and is given by:

$$\text{DICE}(A,B) = \frac{2|A \cap B|}{|A| + |B|}. \tag{4.4}$$

The DICE coefficient penalizes both false positives and false negatives, making it useful for measuring total performance.

Hausdorff Distance The Hausdorff distance has lower bound of 0 (perfect match) with no upper point. This metric measures the maximum minimum distance between two sets of points and is given by:

$$d_H(A,B) = \max\{\sup_{a \in A} \inf_{b \in B} d(a,b), \sup_{b \in B} \inf_{a \in B} d(a,b)\}, \tag{4.5}$$

where $d(a,b)$ is a distance function, for which we use the euclidean norm $d(a,b) = ||a-b||_2$. Hausdorff distance is used to capture the worst errors of each classifier.

Average Symmetric Surface Distance The average symmetric surface distance computes the average minimal distance between the surface points of two sets A and B. Let $\tilde{A}$ and $\tilde{B}$ be sets of the surface points in A and B respectively. The ASSD is then:

$$\text{ASSD}(A,B) = \frac{1}{|\tilde{A}| + |\tilde{B}|} \left(\sum_{a \in \tilde{A}} \min_{b \in \tilde{B}} d(a,b) + \sum_{b \in \tilde{B}} \min_{a \in \tilde{A}} d(b,a) \right), \tag{4.6}$$

where d is again the Euclidean norm. A perfect match between $\tilde{A}$ and $\tilde{B}$ result in an ASSD score of 0. Qualitatively, the ASSD is more representative of the global similarity between two sets than the Hausdorff distance.

4.5.2 Implementation

In this section, we outline the implementation specifics of our DeepLofting pipeline and segmentation classifiers. In particular, we describe the specific methodologies and libraries used for each baseline method, extracting and lofting contours as well as procedures for training CNN networks.

Active Contour Baselines

We compare our DeepLofting pipeline to recent level-set methods used in standard medical image segmentation. Specifically, we compare to the Distance Regularized Level Set (DRLS) [LXGF10] which has been extensively used in the area of medical image segmentation. Additionally, it is common to pre-process images using a vessel enhanced filter before segmentation and we investigate this strategy by using optimally oriented flux (OOF) [Law08] as an image enhancement step prior applying the DRLS for segmentation.

The use of the Distance Regularized Level Set is denoted by DRLS in our results.

Where DRLS is used with OOF pre-processing, it is denoted as DRLS+OOF. During evaluation, DRLS and DRLS+OOF are used as drop-in replacements for 2D segmentation in our DeepLofing pipeline. First, the methods are applied to 2D vessel image patches extracted along vessel path lines to generate 2D segmentations and vessel boundaries. Second, 2D vessel boundaries of each vessel are lofted together to form the final 3D vessel surface. Finally, vessel surfaces are then joined though Boolean operations to form the final 3D patient-specific model. DRLS and OOF have a number of algorithmic parameters which were selected based on performance on the training and validation sets wih reference to the guidelines suggested in [LXGF10]. These parameters used are listed in Table 4.1.

We used an implementation of DRLS obtained from the HistomicsTK [HGM+17] library, and OOF was performed using the Tubular Geodesics library [BTF17] which is based on the Insight Segmentation and Registration Toolkit (ITK) [YAL+02].

Table 4.1. Algorithmic parameters used for DRLS and OOF comparisons

Symbol	Value	Description
$N_{iter,drls}$	100	Number of level-set iterations
λ_{drls}	2.0	DRLS length regularization coefficient
μ_{drls}	0.2	DRLS Energy regularization coefficient
σ_{drls}	0.5	DRLS Gaussian smoothing parameter
α_{drls}	0.9	DRLS Area energy function coefficient
$\sigma_{min,OOF}$	0.1	Minimum vessel scale for OOF enhancement
$\sigma_{max,OOF}$	3.0	Maximum vessel scale for OOF enhancement
$N_{scales,OOF}$	9	Number of scales between $\sigma_{min,OOF}$ and $\sigma_{max,OOF}$ for OOF enhancement

CNN Implementation and Training

As the proposed method builds on existing FCN networks, we compare our I2I-FC architecture to the base I2I network without spatial context processing which is an a 2D adaptation of I2I-3D [MMKT16]. We train both I2I and I2I-FC individually and end-to-end using identical procedures using TF Weights were initialized by sampling from a zero-mean

normal distribution with fixed variance for all weights. Each network was trained with a constant base-learning rate of $1e-4$ and a batch size of 16 for $40,000$ iterations. L_2 regularization with a small coefficient value was added to the binary cross-entropy loss function. Given the difference in data normalization methods (as described in Section 4.4.1), we trained separate CNN classifiers on CT and MR data. Hyper-parameters for networks trained on CT and MR data were kept the same and are summarized in in Table 4.2.

We also compare to the original I2I-3D network [MMKT16] which was re-trained for vessel segmentation on 3D volumes. As with the 2D CNN classifiers, two I2I-3D networks were trained, one on CT data and another on MR. Again, both classifiers were trained using the same hyper-parameters. 2D segmentations were extracted from the I2I-3D volumetric predictions at the same path-points at as image patches After 2D segmentations were captured, they were converted into 2D contours and lofted in the same way as all other methods. We found that this step improves results by ignoring excessive false positives.

All CNN networks were implemented in the Tensorflow software package [AAB+15]. We used the marching squares algorithm from the Visualization Toolkit (VTK) [SML06] for contour extraction.

Table 4.2. Hyper-parameters used for training I2I-FC and I2I networks

Symbol	Value	Description
α	1e-4	learning rate
N_{iter}	$40,000$	Number of training iterations
N_{batch}	16	Training batch-size
W_{init}	7e-2	Weight initialization standard deviation
λ	1e-4	L_2 regularization coefficient
w	128	Width of input images
h	128	Height of input images
I_{CT}	3000	CT pixel value normalization constant

Model Construction

The final stages of our DeepLofting pipeline were carried out in the open source package, SimVascular [UWM+13]. Besides its extensive hemodynamics simulation capabilities, SimVascular provides an interface with which to reorient 2D segmentations around a pathline and to interpolate vessel surfaces. SimVascular also provides specialized Boolean operations for vessel surfaces [UWS16] which were used to form the final 3D models.

4.6 Results

Table 4.3. Comparison between 2D vessel boundaries produced by our methods and baselines split based on imaging modality as well as overall scores. Marching squares was used to extract contours for all methods.

	DICE			Hausdorff			ASSD		
	CT	MR	Overall	CT	MR	Overall	CT	MR	Overall
OOF+DRLS	0.200	0.143	0.176	0.688	0.825	0.728	0.287	0.524	0.357
DRLS	0.268	0.145	0.203	0.667	0.860	0.726	0.215	0.589	0.328
I2I-3D	0.434	0.470	0.442	0.647	0.470	0.589	0.311	0.191	0.272
I2I	0.399	0.601	0.421	0.806	0.359	0.698	0.251	0.132	0.222
I2I-FC (ours)	**0.579**	**0.623**	**0.586**	**0.373**	**0.250**	**0.339**	**0.126**	**0.093**	**0.117**

4.6.1 Segmentation and Contour Results

We begin by discussing the results for 2D segmentation with all methods. We compare results of our I2I-FC segmentation classifier to several common and state-of-the-art methods for segmentation, including the same architecture without the spatial context processing unit and the I2I-3D network. I2I-3D was re-trained following the steps outlined in [MMKT16] for 3D-segmentation and integrated with DeepLofting by extracting planar segmentations from the volume segmentation along the path-lines. We also compare against a distance regularized active contour method, denoted by DRLS [LXGF10] used on raw image data as well as the same active contour with input that have been enhanced enhanced by optimally oriented flux

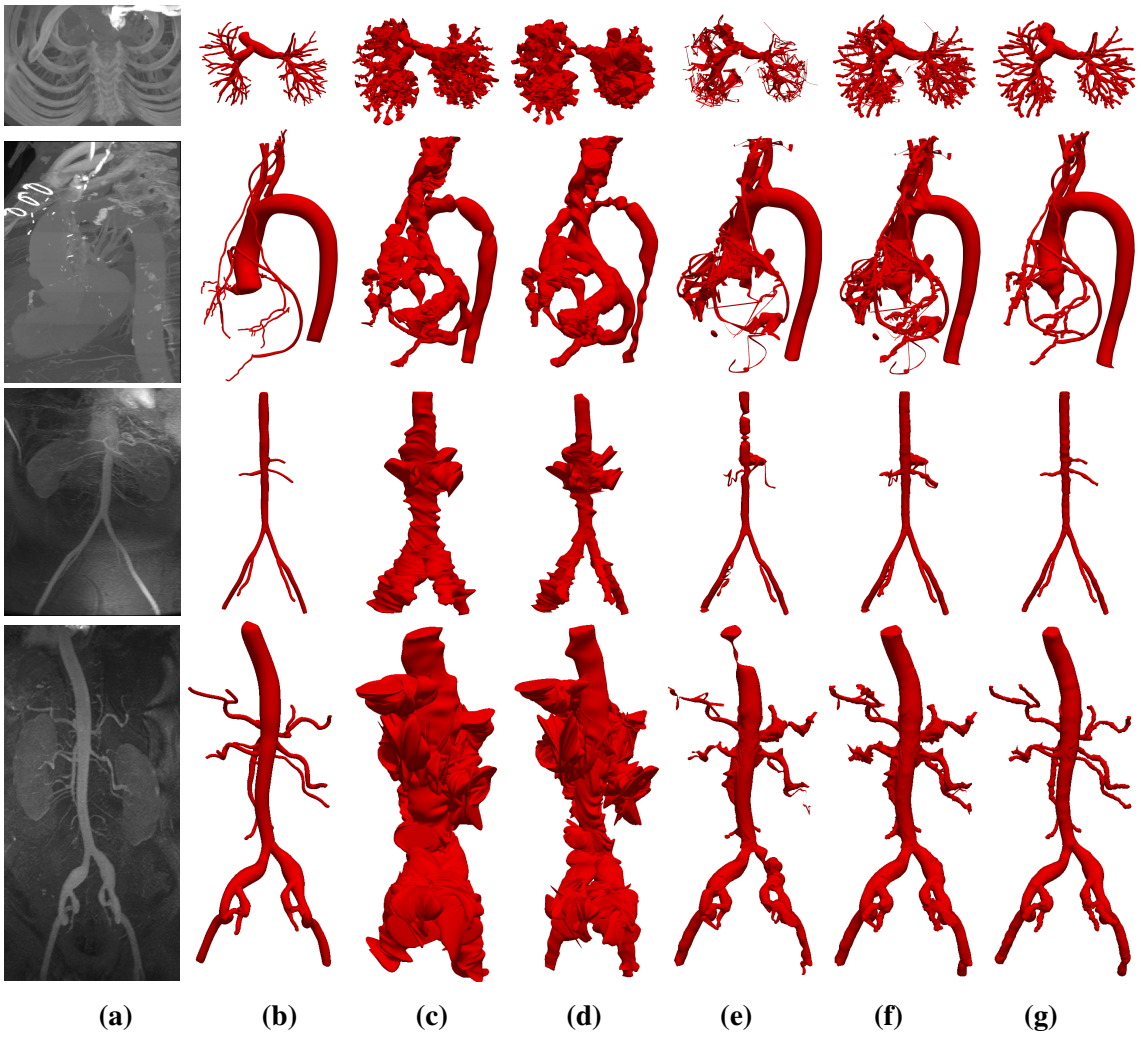


Figure 4.8. Final 3D model results from using different segmentation classifiers. Each row represents a different volume. Each column depicts a result from a classifier: (a) maximum intensity projection, (b) ground truth, (c) DRLS, (d) DRLS+OOF, (e) I2I-3D, (f) I2I, (g) I2I-FC (ours).

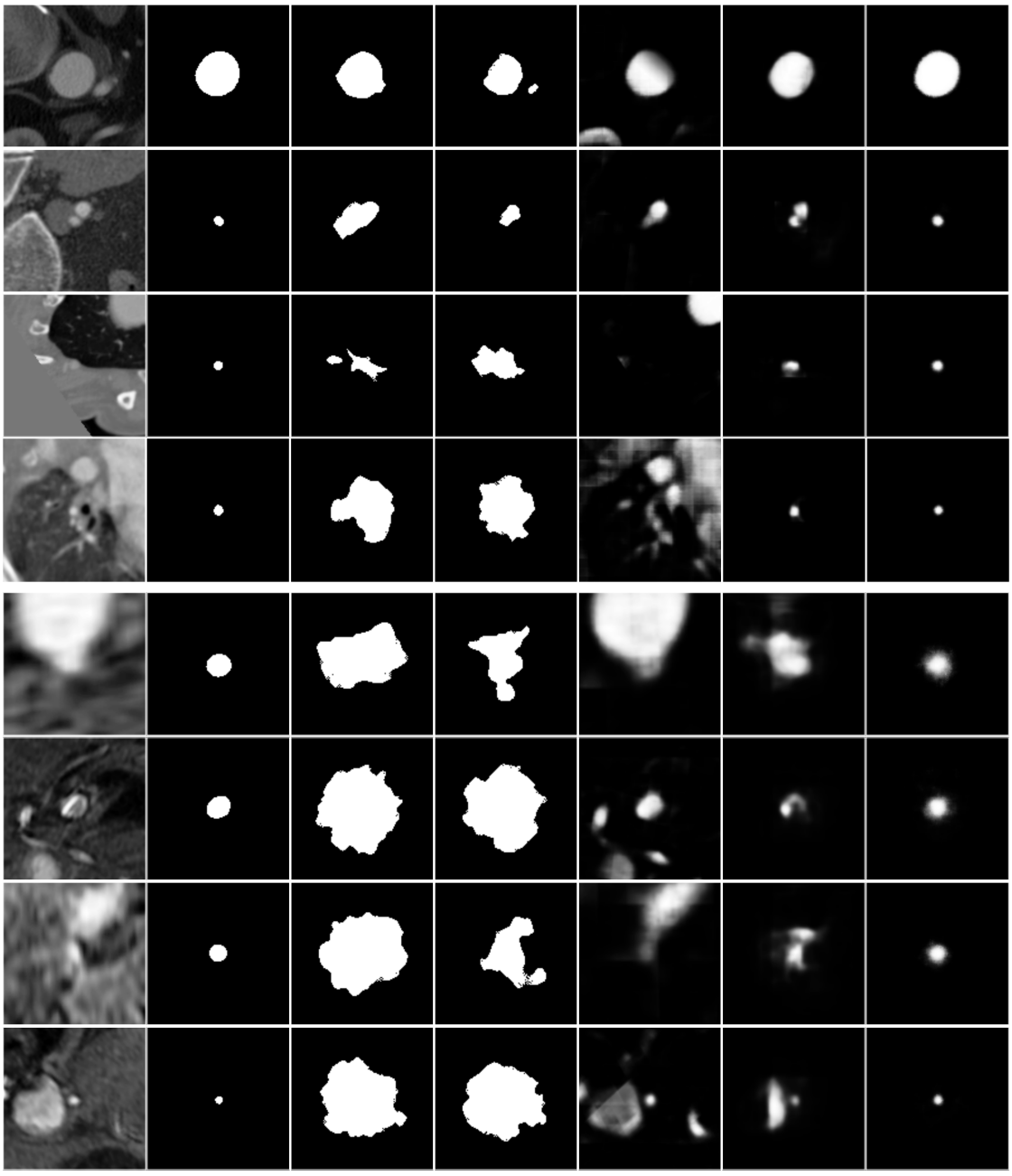


Figure 4.9. Example cross-section segmentation results. The columns represent: (a) image, (b) ground truth, (c) DRLS, (d) DRLS+OOF, (e) I2I-3D, (f) I2I, (g) I2I-FC (ours). The first four rows show results on CT data and second four show results on MR data.

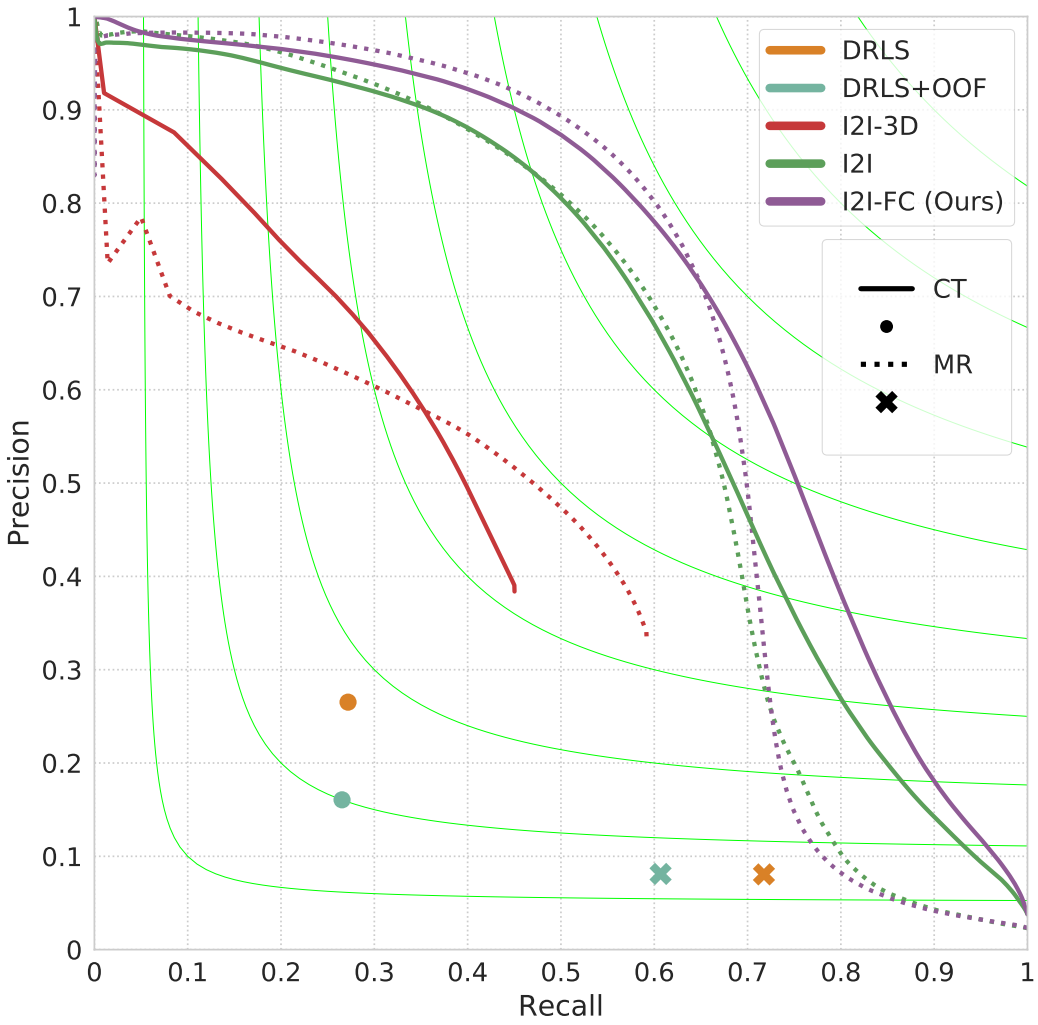


Figure 4.10. Precision-Recall curves for 2D vessel boundaries generated by all methods tested on the test set. CT results appear as the solid lines or a dot and MR performance is denoted with dotted lines or an X. I2I-FC: I2I convolutional network enhanced with our proposed spatial context processor, I2I: fully-convolutional multi-resolution neural network trained for 2D vessel segmentation, I2I-3D: fully-convolutional multi-resolution neural network from [MMKT16], now trained for 3D vessel segmentation. DRLS: Distance regularized level set [LXGF10], OOF+DRLS: DRLS on images enhanced with optimally oriented flux pre-processing [Law08]. The inclusion of spatial context processing (I2I-FC) improves precision-recall on 2D vessel segmentation over standard fully-convolutional networks (I2I, I2I-3D) and classical algorithms such as level sets (DRLS, OOF+DRLS).

[Law08] denoted as DRLS+OOF.

Figure 4.10 depicts precision-recall curves of our MR and CT classifiers which were

generated using a threshold sweep of the predicted segmentations. A summary of classifier performance on CT and MR data as well as the overall performance appears in Table 4.3. Reported DICE scores were computed across the entire dataset, whereas Hausdorff distance and ASSD were computed on a per-contour basis then averaged. We evaluate these metrics at every position along the interpolated pathlines where a ground truth contour is available. The results in Table 4.3 use a single threshold value which match the iso value used for marching squares.

We observe from Figure 4.10 that I2I-FC has improved precision and recall over all other presented 2D segmentation methods. We also tee that I2I-FC has greater performance accounts to the metrics summaries in Table 4.3 which shows improvements of 39%, 49% and 53% for overall DICE, Hausdorff and ASSD metrics when comparing I2I-FC and I2I. In particular, the smaller Hausdorff distance and ASSD produced by I2I-FC, compared to I2I, implies that I2I-FC provides more consistent vessel segmentations, both on average and in terms of worst-case error. Similar improvements hold when comparing I2I-FC to I2I-3D and both DRLS methods. DICE Hausdorff and ASSD improved more for CT images than for MR images for I2I-FC. For example, comparing I2I-FC to I2I, Hausdorff improvements of 54% and 31% are observed for CT and MR modalities respectively. We hypothesize that the differences in accuracy between CT and MR are due to the lower image resolution of MR images compared to CT, leading to increased ambiguity in ground truth vessel location.

Example 2D predicted planar vessel segmentations appear in Figure 4.9, showing the improved consistency of I2I-FC compared to other methods. We see that level set methods are found to be error-prone even after applying OOF pre-processing to the image. The superior performance of both I2I and I2I-FC compared to I2I-3D provides evidence that utilizing vessel pathline information leads to improved accuracy in cardiovascular segmentation. We theorize that the poor performance exhibited by I2I-3D, as shown in Figure 4.10, results from the many false positives seen in the cross sections examples in Figure 4.9. These false positives

lead to ambiguous contours for model construction. In addition, by comparing the output of I2I and I2I-FC, we see the effect of the spatial processing unit, which refines the segmentation to produce smoother contours and consistently produces only a single vessel response by removing secondary vessels and noise near the center of the image. The refinement indicates that I2I-FC successfully incorporated positional information to overcome this limitation of fully convolutional networks.

4.6.2 Comparison of 3D Patient-Specific Models

Table 4.4. Comparison between 3D patient-specific cardiovascular models produced by our methods and baselines split based on imaging modality.

	DICE			Hausdorff			ASSD		
	CT	MR	Overall	CT	MR	Overall	CT	MR	Overall
DRLS+OOF	0.237	0.131	0.203	0.539	0.551	0.542	0.145	0.117	0.136
DRLS	0.322	0.100	0.222	0.549	0.648	0.580	0.114	0.151	0.125
I2I-3D	0.595	0.549	0.587	0.761	0.641	0.722	0.195	0.126	0.173
I2I	0.637	0.627	0.635	0.413	0.359	0.396	0.071	0.067	0.069
I2I-FC	**0.682**	**0.654**	**0.677**	**0.333**	**0.316**	**0.327**	**0.059**	**0.063**	**0.061**

Next, we discuss the results of the full DeepLofting pipeline. The classifiers denoted in Table 4.4 and Figure 4.8 refer to the classifier used to segment vessel cross sections as part of the DeepLofting pipeline. Performance metrics are calculated with binarized volumes of ground truth models and predicted 3D models. As described in Section 4.3.4, each vessel is generated individually then merged through a Boolean operation. Pathlines often extend past the ground truth models, to avoid over-segmentation in these locations, only contours along the length of the vessel surface were lofted. In Table 4.4, we report summary statistics of our method, I2I-FC and other baseline classifiers. DICE scores are calculated by generating hit and miss counts on the entire dataset; ASSD and Hausdorff distances represents the average distance on a per-volume basis. As with 2D performance metrics, we report results for our CT and MR classifiers, as well as the overall performance on the entire data set, obtained by

merging the results from the CT and MR evaluations. All metrics are calculated on binary versions of the 3D models.

From Table 4.4, we see that I2I-FC outperforms all other methods, resulting in improvements of 7%, 18% and 12% for overall DICE, Hausdorff and ASSD metrics. Again we see larger improvements for CT images than for MR images. When comparing 2D results (Table 4.3) and 3D model construction results (Table 4.4), we see an overall improvement in performance which we believe is due reduces false positives after contour extraction with marching squares. Nevertheless, we still see substantial improvements when comparing our CNN (I2I-FC) to the other methods. When comparing our 2D CNN classifiers (I2I and I2I-FC) to I2I-3D, again, we see the benefit of training and prediction using pathline information as they both show considerable improvement over the 3D CNN. Also, as seen when comparing I2I and I2I-FC, we observe consistently superior performance when applying the spatial processing unit.

Figure 4.8 shows example 3D model results, generated using each classifier with the DeepLofting pipeline as well as the ground truth 3D model. We notice the the particularly large distortion and errors when using the level set methods, with and without OOF pre-processing (columns 4.8(c) and 4.8(d)). Many of the contours are malformed and segmentation is inconsistent at each point, leading to a large amount of noise through the model. Though much more consistent than the level set methods, the results of I2I-3D (column 4.8(e)) remain noisy. With this classifier, the contours are erratic, likely due to the over-segmentation exhibited by the underlying classier. In column 4.8(f), we see the results of I2I, the 2D CNN classifier without spatial processing. Here, we see many contours that are too large, most noticeably in the smaller vessels. This classifier fails to ignore extraneous contours when many exist close together which cause errors within these clusters. Finally, in column 4.8(g) we see the results of our full DeepLofting pipeline with the I2I-FC classifier, and we notice the effect of the localized predictions on the final model. I2I-FC produces only a single

segmentation, located correctly along the path-line and we see well-formed vessels even in areas where many vessels overlap. The vessels are consistent in size and shape, leading to a more complete final model.

4.7 Conclusion

We proposed a neural network architecture that substantially improved segmentation performance of FCNs when spatial context in images was essential for making correct predictions. We developed a novel neural network, I2I-FC, that extends I2I with image-wide context processing. Our proposed architectures demonstrate that utilizing this context is essential to producing accurately localized segmentations and generalizes across neural networks and is not restricted to a particular architecture. We combined our architectures with a pathline-based lofting 3D model construction work-flow to form DeepLofting, a new and more efficient method for constructing 3D cardiovascular models. DeepLofting produces models with higher quality than state-of-the-art 2D and 3D methods and baseline active contour methods. DeepLofting automates contour generation, significantly reducing user effort in 3D model construction.

Acknowledgements

Chapter 4, in part, is currently being prepared for submission for publication of the material as it may appear in Gabriel Maher[2], Jameson Merkow[1], David Kriegman, and Alison Marsden. The dissertation author was one of two equal contributing authors of this paper in both algorithm and manuscript development.

[2]Equal contribution

Chapter 5

Conclusion

5.1 Summary of Contributions

This thesis presents three approaches to dense pixel level labeling and demonstrates their effectiveness for segmentation and boundary detection in images and volumetric data.

First, a 3D extension of the popular structured forest classifier for cardiovascular vessel wall localization is introduced. This classifier utilizes an intelligent sampling scheme to train a structured forest classifier to predict patch-to-patch pixel level predictions from domain specific image features which include an adaptive prior. Evaluation of this classifier was carried out on a publicly available cardiovascular dataset where it achieved top performance compared to multiple baselines and similar classifier without these contributions. Furthermore, we demonstrated that this approach is robust to user error by analyzing its predictive capability when error is introduced to the user-supplied *a-priori information*.

Next, two new 3D CNN classifiers for volume and image segmentation are introduced. The first is an extension of the popular HED classifier, extended for use in 3D medical images. The second is a new classifier, I2I, that precisely localizes boundaries and small structures using a novel fine-to-fine, multi-scale architecture. These classifiers were applied to multiple medical imaging applications and demonstrate its effectiveness for natural images. I2I was shown to out-perform current state-of-the-art methods as well as alternative multi-scale

merging strategies on multiple tasks across multiple datasets.

Last, I2I, introduced in Chapter 3, is used as part of a novel cardiovascular model building process, DeepLofting. DeepLofting describes a principled approach to cardiovascular model building that significantly improves efficiency and reduces the need to for human intervention. DeepLofting uses an extensions of the I2I architecture which adds spatial context processing, allowing the classifier to utilize powerful *a-priori* information. We demonstrate the effectiveness of DeepLofting on a dataset of 100 cardiovascular models by reconstructing accurate 3D geometries from both CT and MR volumes.

5.2 Conclusions and Future Directions

The methodologies in this work constitute a step forward in pixel level labeling. In particular, the precise fine-to-fine architecture presented in Chapter 3 has potential use in many applications which require precise localization. The presented techniques are currently used in image-to-image and volume for volume labeling mainly for cardiovascular boundary detection and cerebral segmentation, however, the same methodologies could be used for a wide variety of applications.

Possible 3D application include action recognition in video, as well as broader usage in medical imaging tasks such as breast lesion segmentation, bone fracture detection and any other task where precise localization of small structures is critical. As we show in Chapter 3 and 4, I2I is effective for segmentation in not only 3D volumes but 2D images. With this in mind, the same methodology could be used in 2D natural image applications requiring precise localization such as aerial photography and image disparity detection.

In Chapter 4, spatial context processing is added to the I2I architecture, leveraging the powerful spatial cues prevalent in medical imaging for classification of vessel walls. This property could aid in tasks such as eye/pupil tracking, lip reading, autonomous vehicle sensing, and many others.

The DeepLofting framework could be extended in various ways. DeepLofting relies heavily on user supplied path-lines. A re-framing of this framework could remove this requirement by simultaneously segmenting contours and predicting pathline prediction thereby the entirety of automating the model construction process. In addition, the clinical impact of DeepLofting is yet to be assessed. Though work on measuring this impact is underway, there remains many unanswered questions. Though DeepLofting produces accurate 3D segmentations, are they accurate enough for blood flow simulation, and in what ways to they vary from user generate models. In fact, very little work has been done on characterizing user-variability within the models themselves. Accounting for this variation, is a promising next step for deploying DeepLofting in a wide-spread setting.

The above only represent a personal vision of the directions which could be followed in the near future. However, whatever scenarios arise, precise pixel labeling will play an increasingly important role in both medical image analysis and natural image computing.

Bibliography

[AAB+15] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[AEB06] Michal Aharon, Michael Elad, and Alfred Bruckstein. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on signal processing*, 54(11):4311–4322, 2006.

[AF97] Mohamed N Ahmed and Aly A Farag. Volume segmentation of ct/mri images using multiscale features, self-organizing principal components analysis (sopca), and self-organizing feature map (sofm). In *Proc. of the ICNN97*, volume 3, pages 1373–1378, 1997.

[AMFM11] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5), May 2011.

[APB+08] L Antiga, M Piccinelli, L Botti, B Ene-Iordache, A Remuzzi, and D A Steinman. An image-based modeling framework for patient-specific computational hemodynamics. *Medical & Biological Engineering & Computing*, 46(11):1097–1112, 2008.

[ARAA+16] Rami Al-Rfou, Guillaume Alain, Amjad Almahairi, Christof Angermueller, Dzmitry Bahdanau, Nicolas Ballas, Frédéric Bastien, Justin

Bayer, Anatoly Belikov, Alexander Belopolsky, Yoshua Bengio, Arnaud Bergeron, James Bergstra, Valentin Bisson, Josh Bleecher Snyder, Nicolas Bouchard, Nicolas Boulanger-Lewandowski, Xavier Bouthillier, Alexandre de Brébisson, Olivier Breuleux, Pierre-Luc Carrier, Kyunghyun Cho, Jan Chorowski, Paul Christiano, Tim Cooijmans, Marc-Alexandre Côté, Myriam Côté, Aaron Courville, Yann N. Dauphin, Olivier Delalleau, Julien Demouth, Guillaume Desjardins, Sander Dieleman, Laurent Dinh, Mélanie Ducoffe, Vincent Dumoulin, Samira Ebrahimi Kahou, Dumitru Erhan, Ziye Fan, Orhan Firat, Mathieu Germain, Xavier Glorot, Ian Goodfellow, Matt Graham, Caglar Gulcehre, Philippe Hamel, Iban Harlouchet, Jean-Philippe Heng, Balázs Hidasi, Sina Honari, Arjun Jain, Sébastien Jean, Kai Jia, Mikhail Korobov, Vivek Kulkarni, Alex Lamb, Pascal Lamblin, Eric Larsen, César Laurent, Sean Lee, Simon Lefrancois, Simon Lemieux, Nicholas Léonard, Zhouhan Lin, Jesse A. Livezey, Cory Lorenz, Jeremiah Lowin, Qianli Ma, Pierre-Antoine Manzagol, Olivier Mastropietro, Robert T. McGibbon, Roland Memisevic, Bart van Merriënboer, Vincent Michalski, Mehdi Mirza, Alberto Orlandi, Christopher Pal, Razvan Pascanu, Mohammad Pezeshki, Colin Raffel, Daniel Renshaw, Matthew Rocklin, Adriana Romero, Markus Roth, Peter Sadowski, John Salvatier, François Savard, Jan Schlüter, John Schulman, Gabriel Schwartz, Iulian Vlad Serban, Dmitriy Serdyuk, Samira Shabanian, Étienne Simon, Sigurd Spieckermann, S. Ramana Subramanyam, Jakub Sygnowski, Jérémie Tanguay, Gijs van Tulder, Joseph Turian, Sebastian Urban, Pascal Vincent, Francesco Visin, Harm de Vries, David Warde-Farley, Dustin J. Webb, Matthew Willson, Kelvin Xu, Lijun Xue, Li Yao, Saizheng Zhang, and Ying Zhang. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688, May 2016.

[Bau09] Bauer, C. and Pock, T. and Sorantin, E. and Bischo, H. and Beichel, R. Segmentation of Interwoven 3d tubular tree structures utilizing shape priors and graph cuts. *Medical Image Analysis*, 14, 2009.

[BBH+16] Olivier Bernard, Johan G Bosch, Brecht Heyde, Martino Alessandrini, Daniel Barbosa, Sorina Camarasu-Pop, Frederic Cervenansky, Sébastien Valette, Oana Mirea, and Michel Bernier. Standardized evaluation system for left ventricular segmentation algorithms in 3d echocardiography. *IEEE transactions on medical imaging*, 35(4):967–977, 2016.

[BC11] F. Benmansour and L.D. Cohen. Tubular Structure Segmentation Based on Minimal Path Method and Anisotropic Enhancement. *International Journal of Computer Vision*, 92, 2011.

[BLPL07] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle. Greedy layer-wise training of deep networks. In *NIPS*, 2007.

[BMHA00] Gloria Bueno, Olivier Musse, Fabrice Heitz, and Jean-Paul Armspach. 3d watershed-based segmentation of internal structures within mr brain images. In *Medical Imaging 2000*, pages 284–293. International Society for Optics and Photonics, 2000.

[BRLF13] Carlos Becker, Roberto Rigamonti, Vincent Lepetit, and Pascal Fua. Supervised Feature Learning for Curvilinear Structure Segmentation. In *Medical Image Computing and Computer-Assisted Intervention MICCAI 2013*, Lecture Notes in Computer Science, pages 526–533. Springer, Berlin, Heidelberg, September 2013.

[BST15] Gedas Bertasius, Jianbo Shi, and Lorenzo Torresani. Deepedge: A multi-scale bifurcated deep network for top-down contour detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[BTF17] F. Benmansour, E. Turetken, and P. Fua. *Tubular Geodesics using Oriented Flux: An ITK Implementation*, 2013 (accessed November 01, 2017). http://hdl.handle.net/10380/3398.

[CALR17] Ozgun Cicek, Ahmed Abdulkadir, Soeren S Lienkamp, and Olaf Ronneberger. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation . *MICCAI*, 2017.

[Can86] John Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698, 1986.

[CDPY99] Stefano Cagnoni, AB Dobrzeniecki, Riccardo Poli, and JC Yanch. Genetic algorithm-based interactive segmentation of 3d medical images. *Image and Vision Computing*, 17(12):881–895, 1999.

[CKF11] R. Collobert, K. Kavukcuoglu, and C. Farabet. Torch7: A matlab-like environment for machine learning. In *BigLearn, NIPS Workshop*, 2011.

[CLL+15] Tianqi Chen, Mu Li, Yutian Li, Min Lin, Naiyan Wang, Minjie Wang, Tianjun Xiao, Bing Xu, Chiyuan Zhang, and Zheng Zhang. Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems. *Neural Information Processing Systems, Workshop on Machine Learning Systems*, 2015.

[CPJ06] Sandeep Chaplot, LM Patnaik, and NR Jagannathan. Classification of magnetic resonance brain images using wavelets as input to support vector machine and neural network. *Biomedical Signal Processing and Control*, 1(1):86–92, 2006.

[CPK+14] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *arXiv:1412.7062*, 2014.

[CSVZ14] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *BMVC*, 2014.

[DCFF13] P. F. Davies, M. Civelek, Y. Fang, and I. Fleming. The atherosusceptible endothelium: endothelial phenotypes in complex haemodynamic shear stress regions in vivo. *Cardiovascular research*, page cvt101, 2013.

[DDS+09] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[DHS11] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

[DJV+13] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *arXiv*, 2013.

[dLDM+96] M. R. de Leval, G. Dubini, F. Migliavacca, H. Jalali, G. Camporini, A. Redington, and R. Pietrabissa. Use of computational fluid dynamics in the design of surgical procedures: Application to the study of competitive flows in cavopulmonary connections. *The Journal of Thoracic and Cardiovascular Surgery*, 111(3):502–513, March 1996.

[DT05] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 886–893, 2005.

[DTB06] Piotr Dollar, Zhuowen Tu, and Serge Belongie. Supervised learning of edges and object boundaries. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1964–1971. IEEE, 2006.

[DTPB09] Piotr Dollár, Zhuowen Tu, Pietro Perona, and Serge Belongie. Integral channel features. In *BMVC*, 2009.

[Duf13] Dufour, A. and Tankyevych, O. and Naegel, B. and Talbot, H. and Ronse, C. and Baruthio, J. and Dokládal, Passat, N. Filtering and segmentation of 3D angiographic data: Advances based on mathematical morphology. *Medical Image Analysis*, 17, 2013.

[DY14] Li Deng and Dong Yu. Deep learning: methods and applications. *Foundations and Trends in Signal Processing*, 7(3–4):197–387, 2014.

[DZ13] Piotr Dollár and C Lawrence Zitnick. Structured forests for fast edge detection. In *ICCV*, pages 1841–1848. IEEE, 2013.

[DZ15] Piotr Dollár and C. Lawrence Zitnick. Fast edge detection using structured forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015.

[dZMFY10] Diane A de Zélicourt, Alison Marsden, Mark A Fogel, and Ajit P Yoganathan. Imaging and patient-specific simulations for the fontan surgery: Current methodologies and clinical applications. *Progress in Pediatric Cardiology*, 30(1):31–44, 2010.

[EMHMoCHAMI15] Mahdi Esmaily-Moghadam, Tain-Yen Hsia, Alison L Marsden, and Modeling of Congenital Hearts Alliance (MOCHA) Investigators. The assisted bidirectional glenn: A novel surgical approach for first-stage single-ventricle heart palliation. *The Journal of thoracic and cardiovascular surgery*, 149(3):699–705, 2015.

[ENYW+02] Issam El-Naqa, Yongyi Yang, Miles N Wernick, Nikolas P Galatsanos, and Robert M Nishikawa. A support vector machine approach for detection of microcalcifications. *IEEE transactions on medical imaging*, 21(12):1552–1563, 2002.

[EVGW+10] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, June 2010.

[FBKC+12] Andriy Fedorov, Reinhard Beichel, Jayashree Kalpathy-Cramer, Julien Finet, Jean-Christophe Fillion-Robin, Sonia Pujol, Christian Bauer, Dominique Jennings, Fiona Fennessy, and Milan Sonka. 3d slicer as an image computing platform for the quantitative imaging network. *Magnetic resonance imaging*, 30(9):1323–1341, 2012.

[FCNL13] Clement Farabet, Camille Couprie, Laurent Najman, and Yann LeCun. Learning hierarchical features for scene labeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1915–1929, 2013.

[Fis36] Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of human genetics*, 7(2):179–188, 1936.

[FNVV98] A.F. Frangi, W.J. Niessen, K.L. Vincken, and M.A. Viergever. Multiscale vessel enhancement filtering. *Medical Image Computing and Computer-Assisted Intervention*, 1998.

[FSB+02] Bruce Fischl, David H Salat, Evelina Busa, Marilyn Albert, Megan Dieterich, Christian Haselgrove, Andre Van Der Kouwe, Ron Killiany, David Kennedy, and Shuna Klaveness. Whole brain segmentation: automated labeling of neuroanatomical structures in the human brain. *Neuron*, 33(3):341–355, 2002.

[FSTD99] Bruce Fischl, Martin I. Sereno, Roger B.H. Tootell, and Anders M. Dale. High-resolution intersubject averaging and a coordinate system for the cortical surface. *Human Brain Mapping*, 8(4):272–284, 1999.

[GBBH96] Peter Gibbs, David L Buckley, Stephen J Blackband, and Anthony Horsman. Tumour volume determination from mr images by morphological segmentation. *Physics in medicine and biology*, 41(11):2437, 1996.

[GDS13] Antonio Giorgio and Nicola De Stefano. Clinical use of brain volumetry. *Journal of Magnetic Resonance Imaging*, 37(1):1–14, 2013.

[GGAM14] Saurabh Gupta, Ross Girshick, Pablo Arbeláez, and Jitendra Malik. Learning rich features from rgb-d images for object detection and segmentation. In *European Conference on Computer Vision*, pages 345–360. Springer, 2014.

[Gir15] Ross Girshick. Fast r-cnn. *arXiv:1504.08083*, 2015.

[GL14] Yaroslav Ganin and Victor Lempitsky. Nˆ 4-fields: Neural network nearest neighbor fields for image transforms. In *Asian Conference on Computer Vision*, pages 536–551. Springer, 2014.

[GMA+04] Vicente Grau, AUJ Mewes, M Alcaniz, Ron Kikinis, and Simon K Warfield. Improved watershed transform for medical image segmentation using prior information. *IEEE transactions on medical imaging*, 23(4):447–458, 2004.

[GMY+12] T. J. Gundert, A. L. Marsden, W. Yang, D. S. Marks, and J. F. LaDisa Jr. Identification of Hemodynamically Optimal Coronary Stent Designs Based on Vessel Caliber. *IEEE Transactions on Biomedical Engineering*, 59(7):1992–2002, July 2012.

[GWFM+13] Ian J. Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron C. Courville, and Yoshua Bengio. Maxout networks. In *ICML*, 2013.

[HAGM14a] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Hypercolumns for object segmentation and fine-grained localization. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[HAGM14b] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Simultaneous detection and segmentation. In *European Conference on Computer Vision*, pages 297–312. Springer, 2014.

[HCC+14] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, and Adam Coates. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*, 2014.

[HCG03] Sean Ho, Heather Cody, and Guido Gerig. Snap: A software package for user-guided geodesic snake segmentation. *Submitted to MICCAI 2003*, 2003.

[HCH+16] Qibin Hou, Ming-Ming Cheng, Xiao-Wei Hu, Ali Borji, Zhuowen Tu, and Philip Torr. Deeply supervised salient object detection with short connections. *arXiv.org*, November 2016.

[HGM+17] B. Helba, D. Gutman, D. Manthey, D.R. Chittajallu, J. Beezeley, L. Cooper, S. Lee, and Z. Mullen. *HistomicsTK*, 2017 (accessed November 01, 2017). https://github.com/DigitalSlideArchive/HistomicsTK.

[HT08] J. D. Humphrey and C. A. Taylor. Intracranial and Abdominal Aortic Aneurysms: Similarities, Differences, and Need for a New Class of Computational Models. *Annual Review of Biomedical Engineering*, 10(1):221–246, 2008.

[HZRS15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[ILTT11] Juan Eugenio Iglesias, Cheng-Yi Liu, Paul M Thompson, and Zhuowen Tu. Robust brain extraction across datasets and comparison with publicly available methods. *IEEE transactions on medical imaging*, 30(9):1617–1634, 2011.

[IS15] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456, 2015.

[JM97] Timothy N Jones and Dimitris N Metaxas. Automated 3d segmentation using deformable models and fuzzy affinity. In *Biennial International Conference on Information Processing in Medical Imaging*, pages 113–126. Springer, 1997.

[JMI15] Hans J Johnson, Matthew M McCormick, and Luis Ibanez. *The ITK Software Guide Book 1: Introduction and Development Guidelines-Volume 1*, 2015.

[JSD+14] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. *preprint arXiv:1408.5093*, 2014.

[KB15] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 2015.

[KBBP+14] B. R. Kwak, M. Bäck, M.-L. Bochaton-Piallat, G. Caligiuri, M. J.A.P. Daemen, P. F. Davies, I. E. Hoefer, P. Holvoet, H. Jo, and R. Krams. Biomechanical factors in atherosclerosis: mechanisms and clinical implications. *European heart journal*, page ehu353, 2014.

[KF16] R. Khlebnikov and C.A. Figueroa. Crimson: Towards a software environment for patient-specific blood flow simulation for diagnosis and treatment. *Clinical Image-Based Procedures. Translational Research in Medical Imaging*, 2016.

[KGPS13] Jan Kretschmer, Christian Godenschwager, Bernhard Preim, and Marc Stamminger. Interactive patient-specific vascular modeling with sweep surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2828–2837, 2013.

[Kim14] Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

[KLN+16] K. Kamnitsas, C. Ledig, V.F.J. Newcombe, J.P. Simpson, A.D. Kane, D.K. Menon, D. Rueckert, and B. Glocker. Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation. *Medical Image Analysis*, 2016.

[KM08] Alexander Klaser and Marcin Marszalek. A spatio-temporal descriptor based on 3d-gradients. *XXXX*, 2008.

[Kri00] Krissian, K. and Malandain, G. and Nicholas, A. Model-Based Detection of Tubular Structures in 3D Images. *Computer Vision and Image Understanding*, 80, 2000.

[KSG09] HB Kekre, Tanuja K Sarode, and Saylee M Gharge. Tumor detection in mammography images using vector quantization technique. *International Journal of Intelligent Information Technology Application*, 2(5):237–242, 2009.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[KSVS17] M. O. Khan, D. A. Steinman, and K. Valen-Sendstad. Non-Newtonian versus numerical rheology: Practical impact of shear-thinning on the prediction of stable and unstable flows in intracranial aneurysms. *International Journal for Numerical Methods in Biomedical Engineering*, 33(7):n/a–n/a, July 2017.

[KUH+16] Jens Kleesiek, Gregor Urban, Alexander Hubert, Daniel Schwarz, Klaus Maier-Hein, Martin Bendszus, and Armin Biller. Deep mri brain extraction: a 3d convolutional neural network for skull stripping. *NeuroImage*, 129:460–469, 2016.

[KYD+17] E Kerrien, A Yureidini, J Dequidt, C Duriez, R Anxionnat, and S Cotin. Blood vessel modeling for interactive simulation of interventional neuroradiology procedures. *Medical Image Analysis*, 35:685–698, 2017.

[LABFL09] David Lesage, Elsa D Angelini, Isabelle Bloch, and Gareth Funka-Lea. A review of 3d vessel lumen segmentation techniques: Models, features and extraction schemes. *Medical image analysis*, 13(6):819–845, 2009.

[Law08] Law, M.W.K. and Chung, A.C.S. Three Dimensional Curvilinear Structure Detection Using Optimally Oriented Flux. *European Conference on Computer Vision*, 2008.

[LBD+89] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[LBD+90] Yann LeCun, Bernhard E Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne E Hubbard, and Lawrence D Jackel. Handwritten digit recognition with a back-propagation network. In *Advances in neural information processing systems*, pages 396–404, 1990.

[LBW+08] Le Lu, Adrian Barbu, Matthias Wolf, Jianming Liang, Marcos Salganicoff, and Dorin Comaniciu. Accurate polyp segmentation for 3d

ct colongraphy using multi-staged probabilistic binary learning and compositional model. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[LC10] M.W.K. Law and A.C.S. Chung. An oriented flux symmetry based active contour model for three dimensional vessel segmentation. *European Conference on Computer Vision*, 2010.

[LCB10] Yann LeCun, Corinna Cortes, and Christopher JC Burges. Mnist handwritten digit database. *AT&T Labs [Online]. Available: http://yann. lecun. com/exdb/mnist*, 2, 2010.

[LHD+11] C. Li, R. Huang, Z. Ding, J. C. Gatenby, D. N. Metaxas, and J. C. Gore. A level set method for image segmentation in the presence of intensity inhomogeneities with application to mri. *IEEE Transactions on Image Processing*, 20(7):2007–2016, 2011.

[LHKU98] Chulhee Lee, Shin Huh, Terence A Ketter, and Michael Unser. Unsupervised connectivity-based thresholding segmentation of midsagittal brain mr images. *Computers in biology and medicine*, 28(3):309–338, 1998.

[LKC+95] Huai-Dong Li, Maria Kallergi, Laurence P Clarke, Vijay K Jain, and Robert A Clark. Markov random field for tumor detection in digital mammography. *IEEE transactions on medical imaging*, 14(3):565–576, 1995.

[LLF+12] Wei Li, Shu Liao, Qianjin Feng, Wufan Chen, and Dinggang Shen. Learning image context for segmentation of the prostate in ct-guided radiotherapy. *Physics in medicine and biology*, 57(5):1283, 2012.

[LLS15] Ian Lenz, Honglak Lee, and Ashutosh Saxena. Deep learning for detecting robotic grasps. *The International Journal of Robotics Research*, 34(4-5):705–724, 2015.

[LM01] Thomas Leung and Jitendra Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International journal of computer vision*, 43(1):29–44, 2001.

[LMB+14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[Low04] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[LSD15] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation ppt. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[LSJ+06] Zhiqiang Lao, Dinggang Shen, Abbas Jawad, Bilge Karacali, Dengfeng Liu, Elias R Melhem, R Nick Bryan, and Christos Davatzikos. Automated segmentation of white matter lesions in 3d brain mr images, using multivariate pattern classification. In *Biomedical Imaging: Nano to Macro, 2006. 3rd IEEE International Symposium on*, pages 307–310. IEEE, 2006.

[LXG+15] Chen-Yu Lee, Saining Xie, Patrick Gallagher, Zhengyou Zhang, and Zhuowen Tu. Deeply-supervised nets. *AISTATS*, 2015.

[LXGF10] C. Li, C. Xu, C. Gui, and M.D. Fox. Distance regularized level set evolution and its application to image segmentation. *IEEE Transactions on Image Processing*, 19, 2010.

[LXLF14] Michael KK Leung, Hui Yuan Xiong, Leo J Lee, and Brendan J Frey. Deep learning of the tissue-regulated splicing code. *Bioinformatics*, 30(12):i121–i129, 2014.

[LY07] H. Li and A. Yezzi. Vessels as 4-D Curves: Global Minimal 4-D Paths to Extract 3-D Tubular Surfaces and Centerlines. *IEEE Transactions on Medical Imaging*, 26, 2007.

[LZD13] Joseph J Lim, C Lawrence Zitnick, and Piotr Dollár. Sketch tokens: A learned mid-level representation for contour and object detection. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3158–3165. IEEE, IEEE, 2013.

[MAI99] A. M. Malek, S. L. Alper, and S. Izumo. Hemodynamic shear stress and its role in atherosclerosis. *Jama*, 282(21):2035–2042, 1999.

[Mar13] Alison L Marsden. Simulation based planning of surgical interventions in pediatric cardiology. *Physics of Fluids (1994-present)*, 25(10):101303, 2013.

[Mar14] Alison L Marsden. Optimization in cardiovascular modeling. *Annual Review of Fluid Mechanics*, 46:519–546, 2014.

[MBLS01] Jitendra Malik, Serge Belongie, Thomas Leung, and Jianbo Shi. Contour and texture analysis for image segmentation. *International journal of computer vision*, 43(1):7–27, 2001.

[MBR+09] Alison L. Marsden, Adam J. Bernstein, V. Mohan Reddy, Shawn C. Shadden, Ryan L. Spilker, Frandics P. Chan, Charles A. Taylor, and Jeffrey A. Feinstein. Evaluation of a novel Y-shaped extracardiac Fontan baffle using computational fluid dynamics. *The Journal of Thoracic and Cardiovascular Surgery*, 137(2):394–403.e2, February 2009.

[MFM04] David R Martin, Charless C Fowlkes, and Jitendra Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(5):530–549, 2004.

[MFTM01] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 2, pages 416–423. IEEE, 2001.

[MKK+16] U. Morbiducci, A. M. Kok, B. R. Kwak, P. H. Stone, D. A. Steinman, and J. J. Wentzel. Atherosclerosis at arterial bifurcations: evidence for the role of haemodynamics and geometry. *Thrombosis and haemostasis*, 115(3):484–492, 2016.

[MKS+13] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

[MMKT16] Jameson Merkow, Alison Marsden, David Kriegman, and Zhuowen Tu. Dense volume-to-volume vascular boundary detection. In *Medical Image Computing and Computer-Assisted Intervention*, pages 371–379. Springer International Publishing, Cham, 2016.

[MP43] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

[MPC+02] Francesco Migliavacca, Lorenza Petrini, Maurizio Colombo, Ferdinando Auricchio, and Riccardo Pietrabissa. Mechanical behavior of coronary stents investigated through the finite element method. *Journal of Biomechanics*, 35(6):803–811, June 2002.

[MPTAVG16] Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Pablo Arbeláez, and Luc Van Gool. Deep retinal image understanding. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 140–148. Springer, 2016.

[MS89] David Mumford and Jayant Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on pure and applied mathematics*, 42(5):577–685, 1989.

[MST10] V. Mohan, G. Sundaramoorthi, and A. Tannenbaum. Tubular Surface Segmentation for Extracting Anatomical Structures From Medical Imagery. *IEEE Transactions on Medical Imaging*, 29, 2010.

[MT96] Tim McInerney and Demetri Terzopoulos. Deformable models in medical image analysis: a survey. *Medical image analysis*, 1(2):91–108, 1996.

[MTKM15] Jameson Merkow, Zhuowen Tu, David Kriegman, and Alison Marsden. Structural edge detection for cardiovascular modeling. In *Medical Image Computing and Computer-Assisted Intervention*, pages 735–742. Springer, 2015.

[MX98] Feng Ma and Shaowei Xia. A multiscale approach to automatic medical image segmentation using self-organizing map. *Journal of Computer Science and Technology*, 13(5):402–409, 1998.

[NC14] Tuan Anh Ngo and Gustavo Carneiro. Fully automated non-rigid segmentation with distance regularized level set evolution initialized and constrained by deep-structured inference. *CVPR*, pages 3118–3125, 2014.

[NOV11] W. Nichols, M. O'Rourke, and C. Vlachopoulos. *McDonald's blood flow in arteries: theoretical, experimental and clinical principles*. CRC Press, 2011.

[NUZ00] L.G. Nyul, J.K. Udupa, and X. Zhang. New variants of a method of MRI scale standardization. *IEEE Transactions on Medical Imaging*, 19, 2000.

[OC89] G Paul Otto and Tony KW Chau. region-growing algorithm for matching of terrain images. *Image and vision computing*, 7(2):83–94, 1989.

[OPM02] Timo Ojala, Matti Pietikainen, and Topi Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 24(7):971–987, 2002.

[OS88] Stanley Osher and James A Sethian. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *Journal of computational physics*, 79(1):12–49, 1988.

[Ots79] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66, 1979.

[Pez16] Pezold, S. and Horvath, A. and Fundana, K. and Tsagkas, C. and Andelova, M. and Weier, K. and Amann, M. and Cattin, P.C. Automatic, Robust, and Globally Optimal Segmentation of Tubular Structures. *Medical Image Computing and Computer Assisted Intervention*, 2016.

[PP99] Dzung L Pham and Jerry L Prince. An adaptive fuzzy c-means algorithm for image segmentation in the presence of intensity inhomogeneities. *Pattern recognition letters*, 20(1):57–68, 1999.

[PPO+96] Scott Pohlman, Kimerly A Powell, Nancy A Obuchowski, William A Chilcote, and Sharon Grundfest-Broniatowski. Quantitative classification of breast tumors in digitized mammograms. *Medical Physics*, 23(8):1337–1345, 1996.

[PT01] Regina Pohle and Klaus D Toennies. Segmentation of medical images using adaptive region growing. In *Proc. SPIE Medical Imaging*, volume 4322, pages 1337–1346, 2001.

[PTW98] D. Parker, C.A Taylor, and K. Wang. Imaged Based 3D Solid Model Construction of Human Arteries for Blood Flow Simulations. *International Conference of the IEEE Engineering in Medicing and Biology Society*, 20, 1998.

[RB12] Xiaofeng Ren and Liefeng Bo. Discriminatively trained sparse code gradients for contour detection. In *Advanced in Neural Information Processing Systems*, 2012.

[RCSF13] Samarth S. Raut, Santanu Chandra, Judy Shum, and Ender A. Finol. The Role of Geometric and Biomechanical Factors in Abdominal Aortic Aneurysm Rupture Risk Assessment. *Annals of Biomedical Engineering*, 41(7):1459–1477, July 2013.

[RFB15] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.

[RKM16] A.B. Ramachandra, A.M. Kahn, and A.L. Marsden. Patient-specific simulations reveal significant differences in mechanical stimuli in venous and arterial coronary grafts. *Journal of Cardiovascular Translational Research*, 9, 2016.

[Rob16] Robben, D. and Türetken, E. and Sunaert, S. and Thijs, V. and Wilms, G. and Fua, P. ad Maes, F. and Suetens, P. Simultaneous segmentation and anatomical labeling of the cerebral vasculature. *Medical Image Analysis*, 32, 2016.

[Ros58] Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[RRM04] Torsten Rohlfing, Daniel B Russakoff, and Calvin R Maurer. Performance-based classifier combination in atlas-based image segmentation using expectation-maximization parameter estimation. *IEEE transactions on medical imaging*, 23(8):983–994, 2004.

[RZE08] Ron Rubinstein, Michael Zibulevsky, and Michael Elad. Efficient implementation of the k-svd algorithm using batch orthogonal matching pursuit. *CS Technion*, page 40, 2008.

[SAB05] Rushin Shojaii, Javad Alirezaie, and Paul Babyn. Automatic lung segmentation in ct images using watershed transform. In *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, volume 2, pages II–1270. IEEE, 2005.

[SC05] Frank Y Shih and Shouxian Cheng. Automatic seeded region growing for color image segmentation. *Image and vision computing*, 23(10):877–886, 2005.

[SDN+11] Y. Shang, R. Deklerck, E. Nyssen, A. Markova, J. de Mey, X. Yang, and K. Sun. Vascular Active Contour for Vessel Tree Segmentation. *IEEE Transactions on Biomedical Engineering*, 58, 2011.

[SEM+11] Habib Samady, Parham Eshtehardi, Michael C. McDaniel, Jin Suo, Saurabh S. Dhawan, Charles Maynard, Lucas H. Timmins, Arshed A. Quyyumi, and Don P. Giddens. Coronary artery wall shear stress is associated with progression and transformation of atherosclerotic plaque and arterial remodeling in patients with coronary artery disease. *Circulation*, 124(7):779–788, 2011.

[Set96] J.A. Sethian. *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. Cambridge monographs on applied and computational mathematics. Cambridge University Press, 1996.

[Shi00] Shi, J. and Malik, J. Normalized Cuts and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22, 2000.

[SHK+14] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014.

[SHW+15] Matthias Schneider, Sven Hirsch, Bruno Weber, Gábor Székely, and Bjoern H Menze. Joint 3-D vessel segmentation and centerline extraction using oblique Hough forests with steerable filters. *Med. Image Analysis*, 19(1):220–249, January 2015.

[SKM+15] Daniele E. Schiavazzi, Ethan O. Kung, Alison L. Marsden, Catriona Baker, Giancarlo Pennati, Tain-Yen Hsia, Anthony Hlavacek, Adam L. Dorfman, and Modeling of Congenital Hearts Alliance (MOCHA) Investigators. Hemodynamic effects of left pulmonary artery stenosis after superior cavopulmonary connection: a patient-specific multiscale modeling study. *The Journal of Thoracic and Cardiovascular Surgery*, 149(3):689–696.e1–3, March 2015.

[SLC+14] Ari Seff, Le Lu, Kevin M Cherry, Holger R Roth, Jiamin Liu, Shijun Wang, Joanne Hoffman, Evrim B Turkbey, and Ronald M Summers. 2d view aggregation for lymph node detection using a shallow hierarchy of linear classifiers. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 544–552. Springer, 2014.

[SLJ+15] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015.

[SLLL02] J.S. Suri, K. Liu, Redenm L., and S. Laxminarayan. A Review on MR Vascular Image Processing: Skeleton Versus Nonskeleton Approaches: Part II. *IEEE Transactions on Information Technology in Biomedicine*, 6, 2002.

[SML06] W. Schroeder, K. Martin, and B. Lorensen. The visualization toolkit (4th ed.). 2006.

[SNA+97] Yoshinobu Sato, Shin Nakajima, Hideki Atsumi, Thomas Koller, Guido Gerig, Shigeyuki Yoshida, and Ron Kikinis. 3d multi-scale line filter for segmentation and visualization of curvilinear structures in medical images. In *CVRMed-MRCAS'97*, pages 213–222. Springer, 1997.

[SP97] H Ross Singleton and Gerald M Pohost. Automatic cardiac mr image segmentation using edge detection by tissue classification in pixel neighborhoods. *Magnetic resonance in medicine*, 37(3):418–424, 1997.

[SSP03] P.Y. Simard, D. Steinkraus, and J.C. Platt. Best practices for convolutional neural networks applied to visual document analysis. *International Conference on Document Analysis and Recognition*, 2003.

[SSV+97] J Sijbers, P Scheunders, M Verhoye, A Van der Linden, D Van Dyck, and E Raman. Watershed-based segmentation of 3d mr data for volume quantization. *Magnetic Resonance Imaging*, 15(6):679–688, 1997.

[SVL14] I. Sutskever, O. Vinyals, and Q. Le. Sequence to sequence learning with neural networks. In *Proc. of NIPS*, 2014.

[SWR+16] Udhayaraj Sivalingam, Michael Wels, Markus Rempfler, Stefan Grosskopf, Michael Suehling, and Bjoern H. Menze. Inner and outer coronary vessel wall segmentation from CCTA using an active contour model with machine learning-based 3d voxel context-aware image force. volume 9785, page 978502. International Society for Optics and Photonics, March 2016.

[SWW+15] Wei Shen, Xinggang Wang, Yan Wang, Xiang Bai, and Zhijiang Zhang. Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection draft version. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[SZ14] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *arXiv:1409.1556*, Sept. 4, 2014.

[TB97] Alain Tremeau and Nathalie Borel. A region growing and merging algorithm to color segmentation. *Pattern recognition*, 30(7):1191–1203, 1997.

[TB10] Zhuowen Tu and Xiang Bai. Auto-context and its application to high-level vision tasks and 3d brain image segmentation. *PAMI*, 32(10):1744–1757, 2010.

[TBT+14] Kenji Takizawa, Yuri Bazilevs, Tayfun E. Tezduyar, Christopher C. Long, Alison L. Marsden, and Kathleen Schjodt. Patient-Specific Cardiovascular Fluid Mechanics Analysis with the ST and ALE-VMS Methods. In *Numerical Simulations of Coupled Problems in Engineering*, Computational Methods in Applied Sciences, pages 71–102. Springer, Cham, 2014. DOI: 10.1007/978-3-319-06136-8_4.

[TF09] C.A Taylor and C.A. Figueroa. Patient-specific Modeling of Cardiovascular Mechanics. *Annual Review of Biomedical Engineering*, 11:109–134, 2009.

[TFM13] C.A Taylor, T.A. Fonte, and J.K. Min. Computational Fluid Dynamics Applied to Cardiac Computed Tomography for Noninvasive Quantification of Fractional Flow Reserve. *Journal of the American College of Cardiology*, 61:2233–2241, 2013.

[TMW+01] Paul M Thompson, Michael S Mega, Roger P Woods, Chris I Zoumalan, Chris J Lindshield, Rebecca E Blanton, Jacob Moussai, Colin J Holmes, Jeffrey L Cummings, and Arthur W Toga. Cortical change in alzheimer's disease detected with a disease-specific population-based brain atlas. *Cerebral Cortex*, 11(1):1–16, 2001.

[TND+08] Zhuowen Tu, Katherine L Narr, Piotr Dollár, Ivo Dinov, Paul M Thompson, and Arthur W Toga. Brain anatomical structure segmentation by hybrid discriminative/generative models. *Medical Imaging*, 27(4):495–508, 2008.

[TRN+06] Duygu Tosun, Maryam E Rettmann, Daniel Q Naiman, Susan M Resnick, Michael A Kraut, and Jerry L Prince. Cortical reconstruction using implicit surface evolution: accuracy and precision analysis. *NeuroImage*, 29(3):838–852, 2006.

[TSR+15] J. Tran, D. Schiavazzi, A. Ramachandra, A. Kahn, and A. L. Marsden. Automated tuning for parameter identification in multi-scale coronary simulations. In *APS Meeting Abstracts*, 2015.

[Tu05] Zhuowen Tu. Probabilistic Boosting-Tree: Learning Discriminative Models for Classification, Recognition, and Clustering. *IEEE International Conference on Computer Vision*, pages 1–8, July 2005.

[Tur13] Turetken, E. and Benmansour, F. and Andres, B. and Pfister, H. and Fua, P. Reconstructing Loopy Curvilinear Structures Using Integer Programming. *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[UWM+13] A. Updegrove, N. Wilson, J. Merkow, H. Lan, A.L. Marsden, and S.C. Shadden. SimVascular: An Open Source Pipeline for Cardiovascular Simulation. *Annals of Biomedical Engineering*, 61:1–17, 2013.

[UWS16] A. Updegrove, N.M. Wilson, and S.C. Shadden. Boolean and smoothing of discrete surfaces. *Advances in Engineering Software*, 95, 2016.

[VC02] Luminita A Vese and Tony F Chan. A multiphase level set framework for image segmentation using the mumford and shah model. *International journal of computer vision*, 50(3):271–293, 2002.

[VJ04] Paul Viola and Michael J Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.

[Wan01] K C Y Wang. *Level set methods for computational prototyping with application to hemodynamic modeling*. PhD thesis, Stanford University, 2001.

[Wan16] Wang, C. and Kagajo, M. and Nakamura, Y. and Oda, M. and Yoshino, Y. and Yamamoto, T. and Mori, K. Precise renal artery segmentation for estimation of renal vascular dominant regions. *Medical Imaging: Image Processing*, 2016.

[WHOF96] Susan Wegner, T Harms, Helmut Oswald, and Eckart Fleck. The watershed transformation on graphs for the segmentation of ct images. In *Pattern Recognition, 1996., Proceedings of the 13th International Conference on*, volume 3, pages 498–502. IEEE, 1996.

[WJV+03] J. J. Wentzel, E. Janssen, J. Vos, J. C.H. Schuurbiers, R. Krams, P. W. Serruys, P. J. de Feyter, and C. J. Slager. Extension of increased atherosclerotic wall thickness into high shear stress regions is associated with loss of compensatory remodeling. *Circulation*, 108(1):17–23, 2003.

[WKN+16] Chenglong Wang, Mitsuru Kagajo, Yoshihiko Nakamura, Masahiro Oda, Yasushi Yoshino, Tokunori Yamamoto, and Kensaku Mori. Precise renal artery segmentation for estimation of renal vascular dominant regions. *Proc.SPIE*, 9784, 2016.

[WLK+11] Xunlei Wu, Vincent Luboz, Karl Krissian, Stephane Cotin, and Steve Dawson. Segmentation and reconstruction of vascular structures for 3D real-time simulation. *Medical Image Analysis*, 15(1):22–34, 2011.

[WLW+15] Qian Wang, Le Lu, Dijia Wu, Noha El-Zehiry, Yefeng Zheng, Dinggang Shen, and Kevin S Zhou. Automatic segmentation of spinal canals in ct images via iterative topology refinement. *IEEE transactions on medical imaging*, 34(8):1694–1704, 2015.

[WOJ13] Nathan M. Wilson, Ana K. Ortiz, and Allison B. Johnson. The Vascular Model Repository: A Public Resource of Medical Imaging Data and Blood Flow Simulation Results. *Journal of Medical Devices*, 7(4), 2013.

[WTT+17] Zhenglun (Alan) Wei, Phillip M. Trusty, Mike Tree, Christopher M. Haggerty, Elaine Tang, Mark Fogel, and Ajit P. Yoganathan. Can time-averaged flow boundary conditions be used to meet the clinical timeline for Fontan surgical planning? *Journal of Biomechanics*, 50(Supplement C):172–179, January 2017.

[XB12] Ren Xiaofeng and Liefeng Bo. Discriminatively trained sparse code gradients for contour detection. In *NIPS*, pages 584–592, 2012.

[XLS11] Shengzhou Xu, Hong Liu, and Enmin Song. Marker-controlled watershed for lesion segmentation in mammograms. *Journal of digital imaging*, 24(5):754–763, 2011.

[XT15] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. *Proceedings of the IEEE International Conference on Computer Vision*, 2015.

[YAL+02] T.S. Yoo, M.J. Ackerman, W.E. Lorensen, W. Schroeder, V. Chalana, S. Aylward, D. Metaxas, and R. Whitaker. Engineering and algorithm design for an image processing api: a technical report on itk – the insight toolkit. *Studies in Health Technology and Informatics*, 85, 2002.

[YPC+05] Paul A Yushkevich, Joseph Piven, Heather Cody, Sean Ho, James C Gee, and Guido Gerig. User-guided level set segmentation of anatomical structures with itk-snap. In *Insight Journal, Special Issue on ISC/NA-MIC/MICCAI Workshop on Open-Source Software*, 2005.

[YPCH+06] Paul A. Yushkevich, Joseph Piven, Heather Cody Hazlett, Rachel Gimpel Smith, Sean Ho, James C. Gee, and Guido Gerig. User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. *Neuroimage*, 31(3):1116–1128, 2006.

[YSD04] Jing Yang, Lawrence H Staib, and James S Duncan. Neighbor-constrained segmentation with level set based 3-d deformable models. *IEEE Transactions on Medical Imaging*, 23(8):940–948, 2004.

[ZBG+07] Y. Zhang, Y. Bazilevs, S. Goswami, C.L. Bajaj, and T.J.R. Hughes. Patient-specific vascular NURBS modeling for isogeometric analysis of blood flow. *Computer Methods in Applied Mechanics and Engineering*, 196, 2007.

[ZD14] C Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *Computer Vision–ECCV 2014*, pages 391–405. Springer, 2014.

[Zei12] Matthew D Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.

[ZJRP+15] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip Torr. Conditional random fields as recurrent neural networks. *arXiv preprint arXiv:1502.03240*, 2015.

[ZLG+11] Yefeng Zheng, Maciej Loziczonek, Bogdan Georgescu, Shaohua Kevin Zhou, Fernando Vega Higuera, and Dorin Comaniciu. Machine learning based vesselness measurement for coronary artery segmentation in cardiac ct volumes. In *Medical Imaging: Image Processing*, page 79621K, 2011.

[ZLG+15] Yefeng Zheng, David Liu, Bogdan Georgescu, Hien Nguyen, and Dorin Comaniciu. 3D Deep Learning for Efficient and Robust Landmark Detection in Volumetric Data. In *Medical Image Computing and Computer-Assisted Intervention*, pages 565–572. Springer, 2015.