SEPARATION AND SEGMENTATION OF THE HEPATIC VASCULATURE IN CT IMAGES

By

Qingyang Shang

Dissertation

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

in

Electrical Engineering

May, 2010

Nashville, Tennessee

Approved:

Benoit M. Dawant

J. Michael Fitzpatrick

Robert L. Galloway

D. Mitchell Wilkes

Zhaohua Ding

To my dear parents, my husband and my daughter

ACKNOWLEDGEMENTS

I am grateful to all of those who made this dissertation possible. First and foremost, I would like to express my sincere gratitude to my advisor, Dr. Benoit Dawant, for his guidance and help during this work. Appreciation is also expressed to other members of my dissertation committee, Dr. Michael Fitzpatrick, Dr. Robert Galloway, Dr. Mitchell Wilkes, and Dr. Zhaohua Ding. They gave me valuable suggestions on the work.

There are a lot more people who helped me during my study. I want to say thank you to Dr. William C. Chapman and Logan Clements who helped me on the CT image acquisition, to Rui Li who answered my countless questions while I was in IDL learning, to Zhujiang Cao, Xia Li, Yong Li, Sylvia Ding, and friends at Pathfinder Therapeutics, Inc, with whom I had helpful discussions. I also want to thank all the other members in the lab: Ning Xu, Sri Pallavaram, Natalie Han, Antong Chen, Anusha Rao, Ramya Balachandran and Jack Noble, for their precious friendship and support.

My family gave me infinite support in the pursuit of the degree. I would like to thank my parents and my husband, Shengteng Hu, for their love and inspiration. I dedicate all my work to them.

Financial support of this work came from NIH (Grant No. R44 CA119502).

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

I.1 Specific Aims

The hepatic vessel system is one of the most complex vessel systems in the human body. Accurate analysis of the liver vascular system based on volumetric medical dataset is becoming more and more important for many medical applications, such as quantitative diagnosis, surgical planning and monitoring of the progression of tumors or vascular diseases.

Vessel segmentation is a pivotal step for morphology and topology analysis of the vascular systems. The goal of this work is to develop robust and accurate methods for the separation and segmentation of hepatic vessel systems.

I.2 Background and Significance

I.2.1 Importance of Vessel Segmentation in Liver

The liver is the largest organ in the human body and weighs approximately 1500 grams [1]. Although it belongs to the digestive system, the liver also plays an important role in the blood circulation. The healthy liver is dark red in color because plenty of blood flows through it. The main vessel systems in the liver are: the portal vein, hepatic vein, hepatic artery and bile ducts. They function as follows: the portal vein drains blood from the digestive system and its associated organ (e.g. spleen, pancreas, and gallbladder)

to the liver; after being cleaned by the liver, this blood, together with the de-oxygenated blood, flows into the inferior vena cava via the hepatic veins; then the inferior vena cava (or IVC) carries the blood and sends it back to the heart to get refreshed. Besides the portal vein, the other vascular system that drains blood into the liver is the hepatic artery, which conducts oxygenated blood from the heart to the liver and the digestive organs. Figure 1 [2] shows the main vessel systems in the human liver.

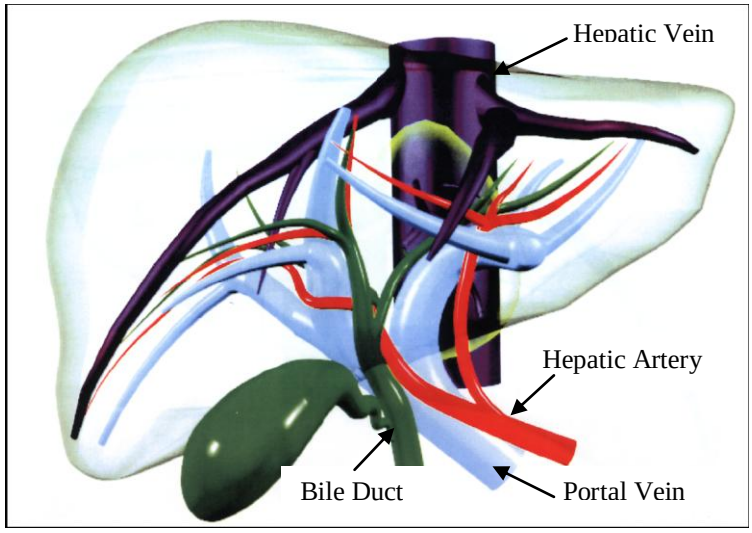


Figure 1. Vessel systems in the liver

Liver cancer is one of the most common carcinomas in the world. Besides being a site for primary cancer (cancer that starts in the liver), the liver is also a common site of metastases (cancer cells that detach from the primary cancer site and travel to other places through lymphatic and/or vascular systems) from a variety of organs such as the lungs, breasts, colon, and rectum. This is because the liver receives blood from surrounding organs via the portal vein. For example, colorectal cancer is a worldwide major health concern. Each year it strikes approximately 850,000 people and accounts for

over 500,000 annual deaths [3]. Data shows that up to 70% of patients with colorectal cancer eventually develop liver metastases [3]. Surgical resection is now a widely accepted treatment for colorectal metastases to the liver (other kind of liver cancer and metastasis as well) since the liver has a property of regeneration. Five-year survival rates are consistently reported between 20% and 35% for patient whose cancer is confined to the liver and is surgically accessible [3]. By contrast, untreated patients with similar disease rarely survived for five years.

Unfortunately, not all patients are resection candidates. Resectability depends largely on the number, size, and location of tumors and their relationship to the main vascular trees. Tumors located too close to important liver blood vessels may be unresectable since certain blood vessels cannot be removed. It must be guaranteed that enough liver tissue supplied by all four vessel systems remains after resection. This requires accurate knowledge of the morphology and structure of the hepatic vasculature.

Living-donor liver transplantation (LDLT) is another effective treatment in patients with small hepatocellular cancers and other rare tumors. It is a procedure in which a piece of the liver is removed from a healthy donor and transplanted into a patient. A precise analysis of the hepatic vascular anatomy is absolutely essential to guarantee donor safety and to predict the postoperative liver function for the patient.

Both liver resection and LDLT require careful surgical planning before the operation. However, a surgeon cannot obtain enough information about the geometrical properties of the vasculature in the liver from just the 2D planar slices in tomographic image volumes. It is critical to provide the surgeon with a 3D patient-dependent vessel

tree model preoperatively. This requires segmentation of the vessels, which is the objective of this work.

I.2.2 Vascular Analysis in Liver

As with many other medical image processing techniques, signal noise, drift in image intensity, and low image contrast make vessel segmentation very difficult. Moreover, the inherent geometrical properties of the hepatic vessel trees, such as thin branches and very small vessels at the end of the vasculature, make the problem even more challenging.

As discussed in the next sections, coverage of the literature shows that a number of methods have been proposed over the years to segment vessels in medical images but only a few have been applied to the liver.

I.2.2.1 General Vessel Segmentation Methods Review

Vessel segmentation algorithms vary depending on the image modality and the application domain. General segmentation methods for vessels or other tubular structures for medical images can be classified into several groups, i.e., threshold-based methods [4]-[8], region growing methods [9]-[13], tracking-based methods [14]-[18], ridge/skeleton-based methods [19]-[21], deformable model based methods [22]-[32], or fuzzy connectedness methods [33]. An excellent review of vessel extraction techniques can be found in [34] and [35].

In general these vessel segmentation methods, although effective for specific applications, do not work well for hepatic vasculature segmentation. For instance, the model-based quantitation method proposed by Frangi et al. [22] models the vessel

segments with a vessel wall surface and a central vessel axis, the latter of which is modeled using a B-spline curve and deformed by minimizing an energy function. This method, and some other methods [15][21], need selection of two ends points to define a geodesic path for initializing the central vessel axis and are suitable for carotid stenosis analysis of a suspicious vessel segment, but is not proper for extracting the whole vessel tree inside the liver. Besides this method, the methods in [23], [25]-[30], and many others, which use level set based approaches or deformable models were proposed for MRA (Magnetic Resonance Angiography) or CTA (Computerized Tomography Angiography) images. But MRA or CTA images are imaging modalities that are designed to image the vessels. The vessel to tissue contrast in these images is thus much higher than it is in the CT images routinely used for liver surgery. For the same reasons, vessel tracking methods, which can produce good results with MRA/CTA images [16][17], cannot produce a continuous and complete hepatic tree in CT images. In [12], Passat et al. used anatomical information to assist in cerebral vasculature segmentation. A vascular atlas was developed to capture the relative position of blood vessels and non-vascular structures of the brain and the head. The atlas was then deformed to the image to be segmented and different thresholds were used in sub-regions to segment the vessels. Although this method allows one to remove more noise and efficiently detect more small vessels, which could not be found with a global threshold, hepatic vessel segmentation cannot benefit from such an atlas-based method since the shapes of both the liver parenchyma and the vessels are highly variable. More importantly, the morphology and branch patterns of the hepatic vasculature are subject to significant variation between individuals. Many other algorithms developed for the segmentation of retinal vessels,

cerebral arteries, aorta or airways can be found in the literature, to the best of our knowledge, none of them has been tested on liver vessels. In fact, most of these algorithms are specifically designed for their application.

I.2.2.2 Hepatic Vessel Segmentation Methods Review

Because of the characteristics of the hepatic vessels, such as individual topological variations, branching pattern complexity, and small size of the vessels at the extremity of the tree, general vessel segmentation methods are ineffectual for hepatic vasculature segmentation. In this section, we review some methods that have been developed specifically for hepatic vessel segmentation.

Soler et al. [36] estimate the intensity distribution of 3 tissue classes: lesions, parenchyma and vessels by fitting 3 Gaussian models to the histogram. The thresholds are defined as crossings of the two adjacent Gaussians. Misclassification can be corrected by morphological closing, distance analysis, thresholding and topological and geometrical constraints.

Glombitza et al. [37] use a histogram entropy algorithm which yields several possible thresholds indicated by peaks in the entropy function; an optimal threshold is selected from these peaks using two fuzzy functions, which describe the relation between the vessels and liver with regard to their volumes and shapes. In practice, these two threshold-based approaches do not delineate the small vessels very well.

Saitoh et al. [38] propose an optimal thresholding method based on structure analysis for hepatic vessel extraction and cancer detection. A temporary threshold is selected near the liver entrance to segment the blood vessels, and then an optimal threshold is determined by varying the temporary threshold and studying the number of

loops produced in the thinned vessels. This method does not distinguish between different vessel systems in the liver. Furthermore, it cannot deal with very small vessels because the threshold is global, nor can it avoid loops in vessel skeletons completely.

Inaoka et al. [39] propose a system, which first extracts candidates of hepatic vessel segments from each 2D image slice using a directional contrast filter, then searches for connecting points according to rules encoding the way vessels extend such as the distance between the connecting points and the direction of vessel segments. Finally the vessels are traced using pre-defined anatomical knowledge based on a tree model; this cuts off all incorrect connections. The first step is tedious and inaccurate. Besides that, the results shown in the paper are very preliminary.

Masutani et al. [40] develop an interactive vessel modeling system for hepatic vasculature for MR images. They use generalized cylinders to represent vessels. The method consists of two steps, the first of which is to extract vessel section data from each slice image using a circle approximation method, an ellipse approximation method, or a minimum width estimation method depending on the vessel orientation relative to the slice. The second step connects vessel sections data based on geometrical relationships between connectable pairs. This method, which has been tested on both phantom and real data sets, failed almost half of the time when dealing with small branches connections. Besides this deficiency, operators need to manually select a point in every vessel section in each slice to start the extraction. This is time consuming and is not feasible for our CT data since small vessels are not easily discernible from the background noise.

Fetita et al. [41] propose an approach for 3D vasculature segmentation in CT hepatic venography based on gray-level mathematical morphology and anatomical

analysis. In this approach, some basic and advanced gray-level morphological operations are performed in a multi-resolution scheme to segment the object of interest from a noisy environment. Information about anatomical features is incorporated in the scheme to detect and insulate the surrounding structures from the vessels. However, separation between the arterial and venous systems was not discussed in this paper.

Shen et al. [42] generate a binary hepatic vessel image by thresholding, then extract the skeleton through analysis of the local maximal voxels in the Euclidean distance mapping of the object. The method trades redundancy of the skeleton voxels for connectedness of the tree by selecting, as the candidate skeleton, voxels of the first three largest in local Euclidean distance which are greater than a threshold. However, the results of the cast liver images show that the method cannot eliminate either of the two effects, skeleton redundancy or disconnection, completely.

Yang et al. [43] present a modeling method to extract and reconstruct portal veins from MR images. Segmentation is achieved by first classifying the voxels into two categories ("blood vessels" and "background") using a Bayesian probability approach and the expectation maximization (EM) algorithm, then isolating the portal vein and its connected veins using an active surface model implemented by the level set method. This modeling method, which was designed to provide geometric boundary conditions for computational fluid dynamics (CFD) simulations of the blood flow inside the portal vein, can only segment and visualize the main portal vein and its connected veins, and small vascular branches segmentation were not discussed.

Hemler et al. [44] proposed a method of segmenting the major vessels in the liver as required for proper radio frequency ablation treatment plan formulation and analysis.

The image is filtered with a 2D median filter and a 2D coherence enhancing diffusion filter to remove intensity variations and make the areas with a similar intensity values more homogeneous. Then a 3D connected component analysis is performed on the maximum intensity projection of the image and a morphological closing operation is followed to fill small gaps and holes. This method, which probably even needs some manual editing, can only segment the very major vessels in the liver that are near the tumor treatment areas. It is not suitable for small vessel segmentation.

Pock et al. [45] propose a tubular structure detection filter, which can provide a radius estimate based on a multiscale medialness function. Based on the filter output, centerlines of the tubes are extracted and the vessel tree is reconstructed. The final segmentation step uses the tube representation to initialize and constrain a level set method for codimension-two geodesic active contours for tubular structures. As is the case for several methods mentioned above, this method cannot deal with vessel branches of higher orders, especially those disconnected with the main vessel trunk because of noise.

Of the various image segmentation methods proposed, region growing has been one of the most popular. This method segments images by first selecting a starting point and then incrementally recruiting pixels to a region based on some criteria.

Zahlten et al. [46] propose a voxel based region growing algorithm with an emphasis on bifurcation detection. Starting from a seed point, the algorithm expands stepwise to trace voxels which satisfy a threshold. The propagation of the algorithm produces a “wave front” which represents the boundary of the currently segmented object. Bifurcations are recognized when one of the connected components of the wave front

splits into two or more parts. However, for image data sets with vessel or liver tissue intensity variations and noise, it is difficult to achieve an accurate result.

Dokladal et al. [47] develop an approach to segment the 3D hepatic vessel system by voxel growing either in the object or the background, with a grey level threshold as the stopping criterion. It differs from other region growing methods in that the growing procedure is restricted to simple points to preserve the homotopology by using priority FIFO lists. The results show that segmentation by background reconstruction is better than segmentation by object construction. But the richness of the vessel structure of both results is sensitive to the stopping criterion.

Selle et al. [48][49] use a threshold-based region-growing method to extract vessels in liver. An optimal threshold is automatically selected based on the fact that the number of the segmented voxels is linearly increasing as the threshold decreases up to a point where there is a sudden change due to the fact that liver tissue is included when the threshold is below the optimal value. Then graph theoretical methods are used to determine vessel skeletons. Liver segments are then approximated based on the branch structures of the portal vein. Finally, vessel visualization is performed by fitting graphics primitives along the skeletons. This method involves some manual interaction and is not fully automatic.

Wan et al. [50][51] propose an algorithm called Seeded Region Growing to extract and represent vascular trees in rats in micro-CT images. They define a set of theoretical criteria for a subclass of region-growing algorithms that are insensitive to the selection of the initial seeds. The growing process consists of three steps: segmenting 1D regions of each row of the image; merging segmented regions of adjacent rows to obtain

region segmentation of each slice; and merging regions of adjacent slices to obtain final result of the 3D image. Separation of different vascular systems and the feasibility for human hepatic vessels segmentation have not been discussed.

Beichel et al. [52] develop a method for portal vein segmentation, the idea behind which is to first enhance the tubular structure by a Hessian filter. Potential vessels are then identified with an iterative region growing method with a successively relaxed criterion. Finally, connected components that lie close to the vessels are reconnected to the main trunk based on the minimum cost path. In this paper the centerline is extracted using a sequential 3D curve thinning algorithm [53] and liver segments are approximated using nearest neighbor approximation method. In this method, labeling liver segments and separating different vessel systems are done manually.

Erdt et al. [54] propose a Hessian based hepatic vessel enhancement filter by designing a vesselness function which is computed from a pre-defined vessel model to avoid parameters adjustment, and an iterative region growing segmentation method with an initialized threshold from the response of the vessel model. Results may need to be refined by manual interaction.

Kaftan et al. [55] propose a two stage venous vasculature segmentation method for CT liver image. In the first stage, the main vessel branches are delineated using globally optimal graph-cuts algorithm, with an initial segmentation which is to determine foreground and background seed points. In the second stage, smaller vessels are detected by a graph based tracking approach based on multi-scale medialness filter.

Freiman et al. [56] develop a variational method for liver vessel segmentation and visualization. To detect bifurcations and complex vessel structures, it integrates Hessian-

based geometrical vesselness measurement and vessel surface properties to form an energy function which is minimized by solving the Euler-Lagrange equation.

Yi et al. [57] use a locally adaptive region growing approach to segment vascular networks in 3D CTA/MRA head/neck/abdominal images. Region growing is restricted to local cubes, and connected component labeling on six faces of the cube is used for detection of bifurcation and estimation of the next cube's position. Tschirren et al. [33] adapt fuzzy connectedness with directional affinity and region of interests (ROI) with cylindrical shape to segment CT airway images. The orientation of new ROI is determined by the skeleton of the current segmentation results. For both methods, locally adaptive analysis in the ROI helps identifying small vessels; but the approach that the vessel segmentation in the local ROI is repeated throughout the whole image is not efficient, especially when such approach is applied to liver, which has complex branching patterns.

To sum up, existing methods for hepatic vessel segmentation have deficiencies in one or more of the following areas: small and/or detached vessel delineation, separation between different vessel systems, need for tedious manual interaction, disconnectedness of the final results, and time efficiency issues. To address such problems, we propose an approach for hepatic vasculature segmentation for CT abdominal images. It can differentiate the portal vein and the hepatic vein based on image registration and histogram analysis, and can identify small vessels, even when pieces are disconnected from the main tree, with an iterative region growing and reconnection method. To reduce the running time and maintain the accuracy of the segmentation results, we use a conventional region growing method for main vessel trunk detection and extract vessel

branches with high orders with an adaptive directional region growing method. It also guarantees that the final segmentation results are connected components.

Hepatic vascular segmentation results are usually binary volumetric data sets. They often have irregularities, such as cavities, holes or bumps, at vascular surfaces. These disturbances without topological meanings are caused by image noise and encumber further geometrical and topological analysis which is necessary for liver surgical planning. A skeletal tree representation is better for identifying branching patterns and determining liver segments supplied by different vascular systems. So topological thinning often follows vessel segmentation [36][37][48][52]. Moreover, in our approach, as in [33], we use the skeleton to guide the region growing process by providing possible directions for vessel extension.

I.2.2.3 Vessel Separation Methods Review

The objective of our study is to segment both the portal vein and hepatic vein. However, due to the imaging protocol, the portal vein and hepatic vein are enhanced simultaneously during the venous phase. So the first step is to separate these two types of veins. In the past, only a limited amount of work has been done to address the issue of vessel separation.

Intravascular contrast agent offers a prolonged blood residence time, allowing acquisition of the steady state images of the arteries and veins with higher contrast and better resolution in contrast-enhanced magnetic resonance angiography (CE-MRA). However, an inevitable disadvantage is the simultaneous enhancement of arteries and veins, which makes the analysis of either vessel system a difficult task. To solve this

problem, separating arteries from veins is essential before any type of vessel analysis can be achieved.

Bemmel et al. [58] introduce a level-set-based method to separate the artery and vein in blood pool agent contrast-enhanced MRA. One of the novelties they claim, which is to use the central arterial/venous axis as initialization for the level set propagation, is also a drawback of the method, because it requires extraction of the central arterial/venous axis before the separation and segmentation of the vessels, which involves user interaction. Moreover, this method is only feasible for the separation of a small number of overlapping arteries and veins, not for the application of a whole vessel tree.

Niessen et al. [59] improve artery visualization by segmenting the major venous structures then suppressing them in maximum intensity projections. Again, this method is limited to the case of a small number of the main overlapping veins, which need to be selected by the user.

Lei et al. [60] reported an artery-vein separation method for MRA using the concept of fuzzy connectedness. They segment the entire vessel structures from the background via absolute fuzzy connectedness first, then separate artery from vein within the whole vessel structures by specifying seed voxels for both vessels and via relative fuzzy connectedness.

Sonka et al. [61] propose a separation method based on graph searching which consists of the following steps: 1) slightly overgrow a vessel tree using seeded region growing; 2) inside the segmented tree, starting from user-defined points, a graph searching similar to dynamic programming is performed to find a centerline of the tree; 3) a constrained morphologic thickening algorithm is used to grow the artery and vein from

the searching result based on vessel labeling propagated to all vessel segments along the centerline.

Bock et al. [62] develop a correlation analysis method to separate the arteries and veins in 3D MRA of the lung based on the difference of the arterial and venous vascular signals in temporal acquisition. About 7 or 8 MRA data sets are acquired within a single breath hold, and the arterial and venous cross-correlation map are computed and analyzed. This method is limited by the acquisition.

Tizon et al. [63] describe an algorithm which uses the gray-sale degree of connectedness to split the original volume into different vessel parts. This region growing method needs the user to interactively define seed regions in the arterial and venous vessels, and it may fail when the contrast between the arterial and venous regions is poor or when an arterial stenosis occurs, due to the fact that the connectedness values associated with the wrong seeds may be higher at an interested voxel.

The approaches mentioned above, besides having their own limitations, all focus on the separation of artery and vein in contrast enhanced MRA images, which have great differences in image contrast and other qualities from our liver CT images used for liver surgical planning. So they cannot produce good results for our application. Instead we propose a vessel separation method with automatic thresholds selection based on an analysis of the 2D intensity histogram.

CHAPTER II

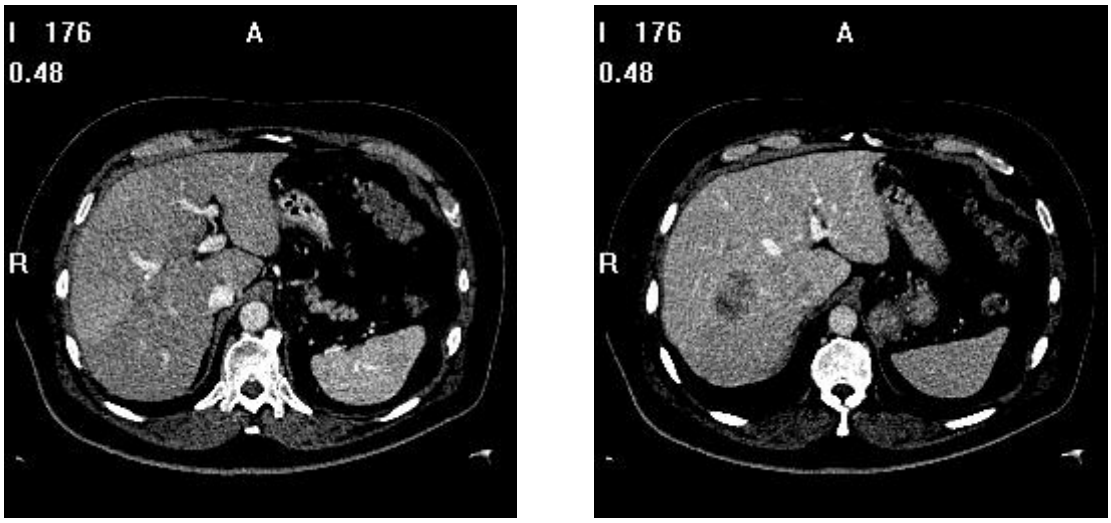
IMAGE PREPROCESSING – IMAGE REGISTRATION AND ENHANCEMENT

II.1 Introduction

II.1.1 Image Data

The image modality that we are interested in is Computerized Tomography (CT). One patient study usually includes several data sets that are acquired with different imaging parameters. These include the time at which the images are acquired after injection of a contrast agent or the slice thickness.

The abdomen of each patient is CT scanned (Sensation 64, SIEMENS Medical System) after injection of a contrast agent. The hepatic vascular systems become visible gradually as the contrast agent diffuses inside the vessels. Different vessel systems are captured at different time based on the perfusion of the contrast agent. The hepatic arteries are the first vessel system that is enhanced with respect to liver tissues, then the contrast agent passes through portal veins and finally flows into the inferior vena cava via the hepatic veins. There are two imaging phases in our study: the arterial phase, and the venous phase. Typically, datasets taken at the arterial phase show the hepatic artery and the portal vein more clearly, while those taken at the venous phase show the portal vein and the hepatic vein better. The following figure shows an example of CT axial images in both phases for the same patient.



(a). Axial image of the arterial phase (b). Axial image of the venous phase

Figure 2. An example of a CT slice of the original image

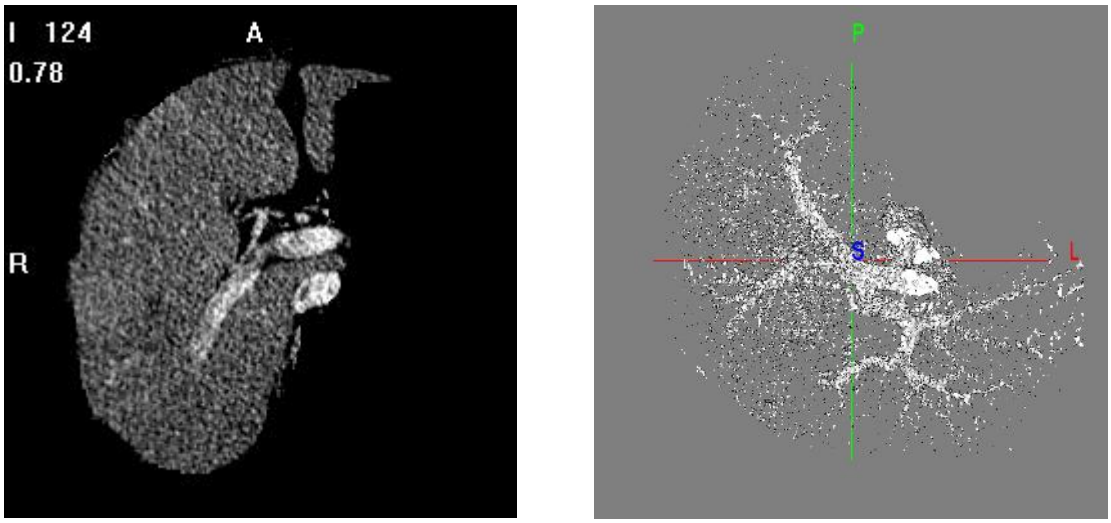
At each imaging phase, the patient is scanned with two thickness parameters, 1mm and 5mm. Images with thick slices (5mm) have higher signal to noise ratio but do not contain enough information for accurate vessel segmentation. On the contrary, images with thin slices (1mm) show more details but the signal to noise ratio is low.

We have used 5 image datasets in this study. Typical image size and voxel size for the 1mm thickness volumes are 512*512*250 and 0.8mm*0.8mm*1mm, respectively, and those for the 5mm thickness volumes are 512*512*50 and 0.8mm*0.8mm*5mm.

II.1.2 Characteristics of the Image Data

Thin slice CT images (1mm) have relatively low contrast-to-noise ratio. Figure 3(a) is an example of a CT slice with a 1mm thickness. Besides the very big vessel branch, there are several smaller vessel cross-sections appearing in this slice. The fact that the edges between the vessels and liver parenchyma are not very clear and that small

vessel cross-sections are not easy to distinguish from the noise makes the segmentation of these vessels a difficult task. Figure 3(b) is a preliminary segmentation result using just a threshold, which gives us some idea of the noise level in the dataset. So, pre-processing such as image enhancement is an important step in the process.



(a). A liver CT slice of thickness 1mm (b). Thresholding result

Figure 3. A CT slice example and its thresholding result

Anatomically, the portal vein lies parallel to the hepatic artery and the bile ducts. It is usually considered as the leading structure for these three vessels and it is based on the portal vein that the liver is divided into eight segments that are used for guidance during liver surgery. Since the hepatic artery is relatively small and has the same orientation as the portal vein, it is neglected in this study. Our ultimate goal is to separate and segment the portal vein and the hepatic vein.

Although the venous phase images are acquired right after the arterial phase, motion artifacts do exist between these two imaging phases because of human

respirations and patient body movements. Thus the portal vein in the arterial phase image and that in the venous phase image are not aligned perfectly. Such displacement is subtle but critical for both vessel separation and segmentation. So registration between the two image phases is essential.

II.2 Image Registration

II.2.1 Methods

To register the source image (the arterial phase image) to the target image (the venous phase image), a two step approach has been used. First the images are registered using a rigid body transformation. This is followed by a non-rigid registration. Both the rigid and non-rigid registration algorithms use a normalized mutual information-based approach. The rigid transformation is computed on the whole CT image at its original size, with an intensity window that covers the intensity range of the image. To reduce execution time for the non-rigid registration step, we define a bounding box on both images, which contains the whole liver, and eliminates other surrounding structures as much as we can. Then we deform the cropped source image to the cropped target image. Instead of using the minimum and maximum intensities of the foreground voxels as the valid intensity ranges of the source and target images as is done in the rigid case, our non-rigid registration scheme computes the joint histogram, and thus the normalized mutual information, within the estimated range of vessel intensities. This is done because the goal of the registration is to align the vascular structures. The liver parenchyma is of less importance and is treated as background in this step. Finally we apply a liver mask to

remove the remaining structures surrounding the liver. The liver mask is obtained by segmenting the liver using a level set-based approach with an accumulative speed function proposed by Cao [64]. The liver segmentation is conducted on the target image (the venous phase image) and the liver mask is also applied to the non-rigid registered arterial phase image.

The non-rigid registration algorithm we have used is the intensity-based adaptive bases algorithm [65]. This approach models the deformation field with compactly supported radial basis functions in a multi-scale and multi-resolution way. Resolution refers to the image spatial resolution, which varies from coarse to fine. Scale is determined by the number and the support region of the basis functions. At a given resolution, the algorithm progresses from a larger scale, which uses few basis functions with large support, to a smaller scale, which uses more basis functions but their support region is reduced. The overall deformation field is modeled as the sum of the deformations occurring at each resolution and scale:

$$v(x) = v_1(x) + v_2(x) + \cdots + v_M(x) \tag{1}$$

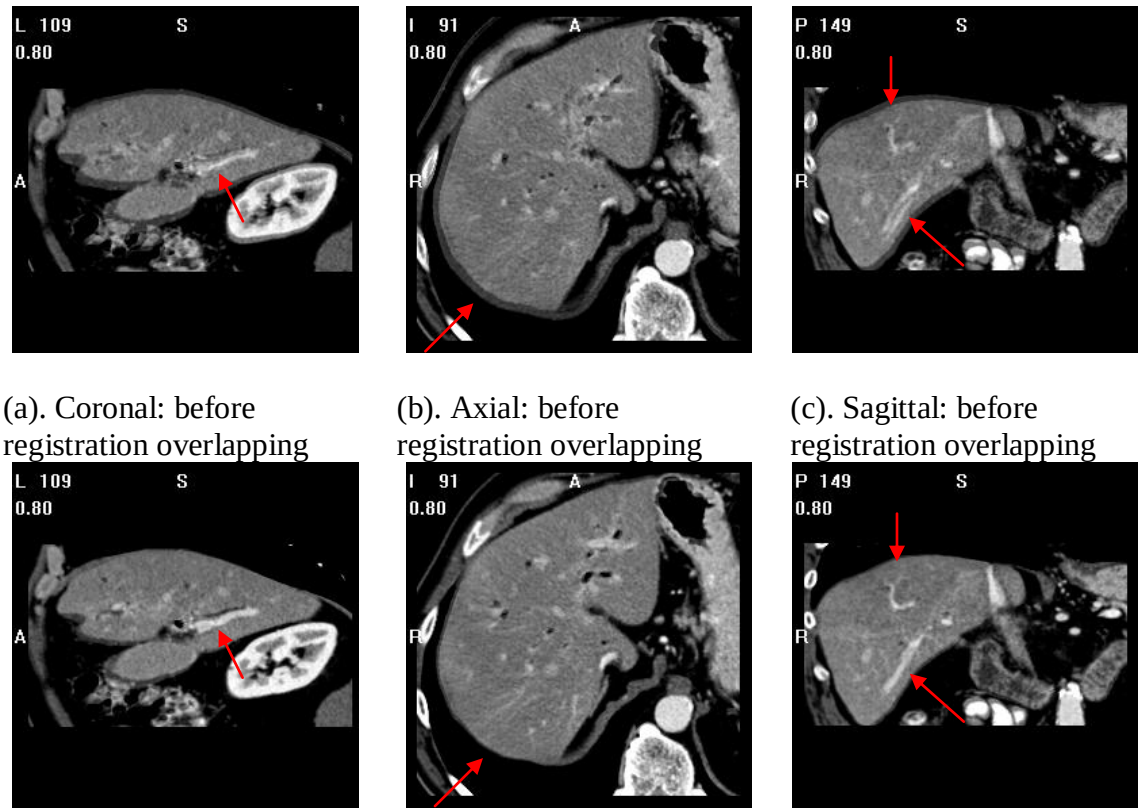
$$v_i(x) = \sum_{j=1}^{N} c_j \Phi(x - x_j) \tag{2}$$

where $v(x)$ is the final deformation field, $v_i(x)$, $i = 1 \cdots M$ is the intermediate deformation field at a certain level i, and M is the total number of levels, which are all the combinations of resolution and scale; Φ is a radial basis function (RBF) centered at voxel point $x_j, j = 1 \cdots N$, $c_j, j = 1 \cdots N$ is the coefficient of the corresponding RBF, and N is the total number of RBFs over the image domain.

II.2.2 Results and Discussion

II.2.2.1 Rigid Registration Results

The arterial phase image is first registered to the venous phase image using a rigid transformation. Figure 4 shows the source images (the arterial phase) before and after rigid registration overlapped on the target images (the venous phase) in coronal, axial, and sagittal views, respectively. The first row shows the original source images overlapped on the target images. The second row shows the rigidly registered images overlapped on the target images. The red arrows show areas of improved alignment between the source and target images. From the results we can see that the rigid transformation registers the liver (and the other surrounding structures, of course) and the big vessels well. But it is not sufficient for small vessels. We will show later (Figure 5 to 9) that non-rigid registration is necessary to align small vessels in the two phases.



(a). Coronal: before registration overlapping

(b). Axial: before registration overlapping

(c). Sagittal: before registration overlapping

(d). Coronal: after registration overlapping

(e). Axial: after registration overlapping

(f). Sagittal: after registration overlapping

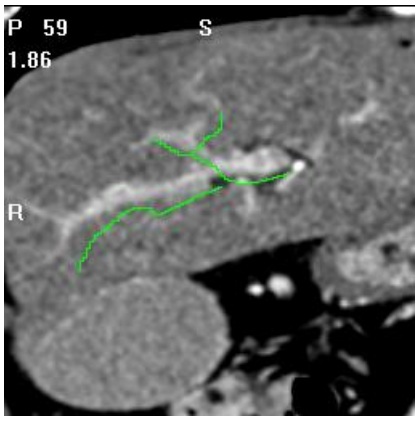
Figure 4. Rigid registration results

II.2.2.2 Non-rigid Registration Results

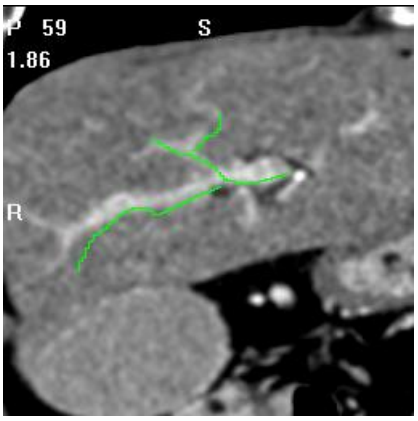
The non-rigid registration scheme approaches the final deformation field iteratively across a number of scales and resolutions. In our experiments, the registration starts from an image down-sampled two levels below the full spatial resolution, and a total number of 15 scales are unevenly distributed between these 3 resolutions: 4 scales for the lowest, 9 scales for the middle, and 2 for the highest resolution. At each scale, registration is confined to areas with intensity above 1000, which is roughly the lower

bound of the vascular structure. Such intensity restriction not only helps to reduce the computation time, but also results in a better alignment between vessels.

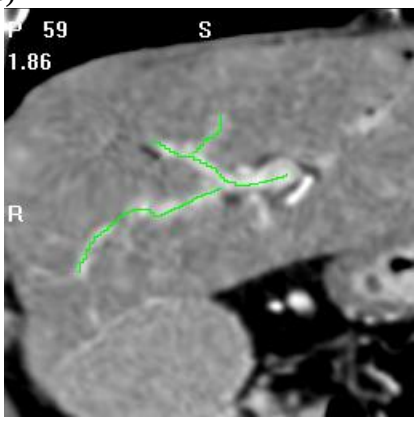
Figure 5 shows a zoomed-in view in a pair of images before and after registration. Figure 5(a) is the original arterial phase image; (b) and (c) are the deformed images after rigid and non-rigid registrations, respectively; and (d) is the venous phase image, to which the arterial phase image is registered. The green curves are the manually drawn centerlines of the vessel in the venous phase image. For comparison, they are also shown on the other images.



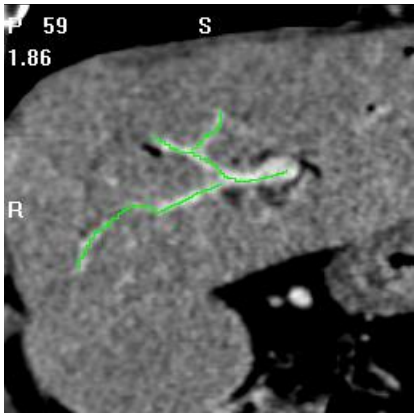
(a). The arterial phase image (source image)



(b). The arterial phase image after rigid registration



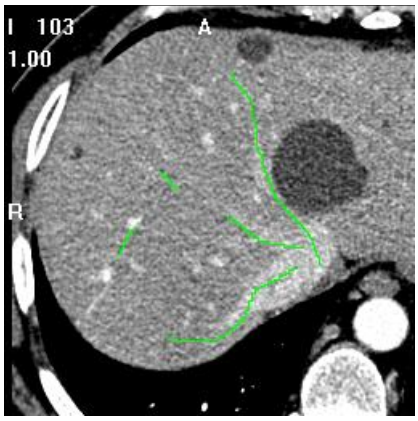
(c). The arterial phase image after non-rigid registration



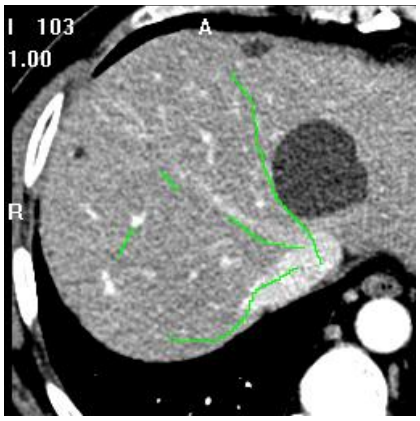
(d). The venous phase image (target image)

Figure 5. Sagittal images with manually-drawn vessel centerlines

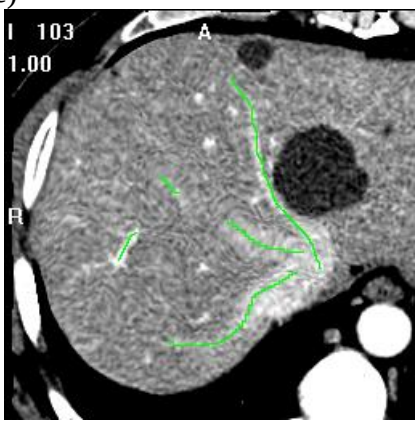
This figure shows that the green curves align very well with the vessels in the non-rigidly registered image (Figure 5(c)), but deviate from the vessels in the rigidly registered image (Figure 5(b)), and of course, deviate from vessels in the original source image (Figure 5(a)), too. Figure 6 and 7 show two other registration examples.



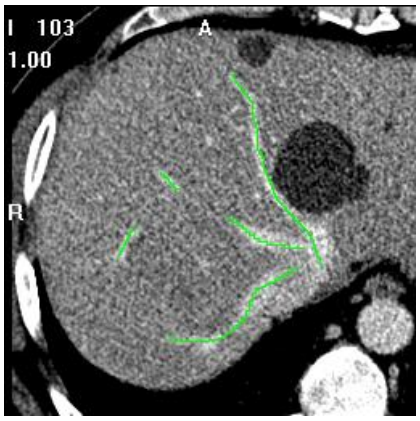
(a). The arterial phase image (source image)



(b). The arterial phase image after rigid registration

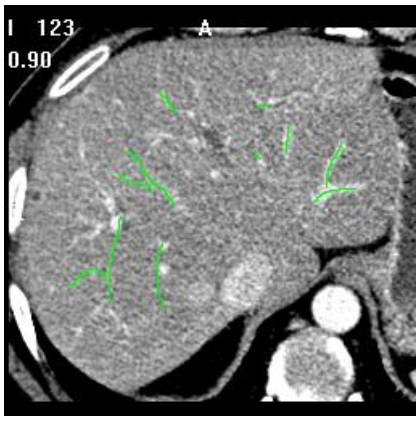


(c). The arterial phase image after non-rigid registration

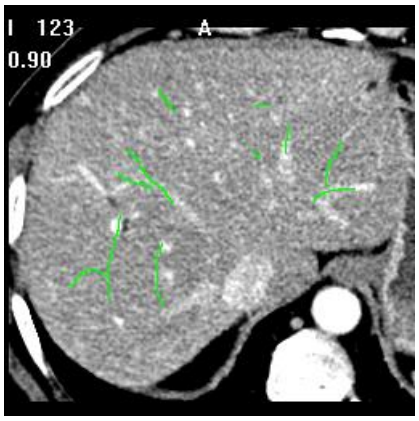


(d). The venous phase image (target image)

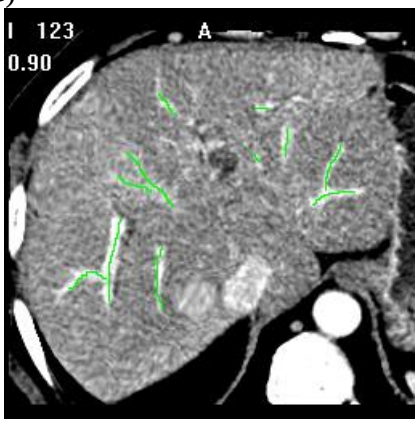
Figure 6. Axial images with manually-drawn vessel centerlines



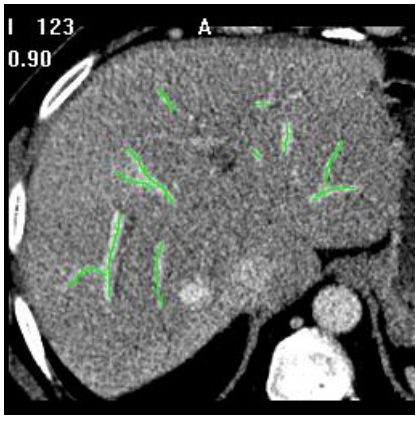
(a). The arterial phase image (source image)



(b). The arterial phase image after rigid registration



(c). The arterial phase image after non-rigid registration

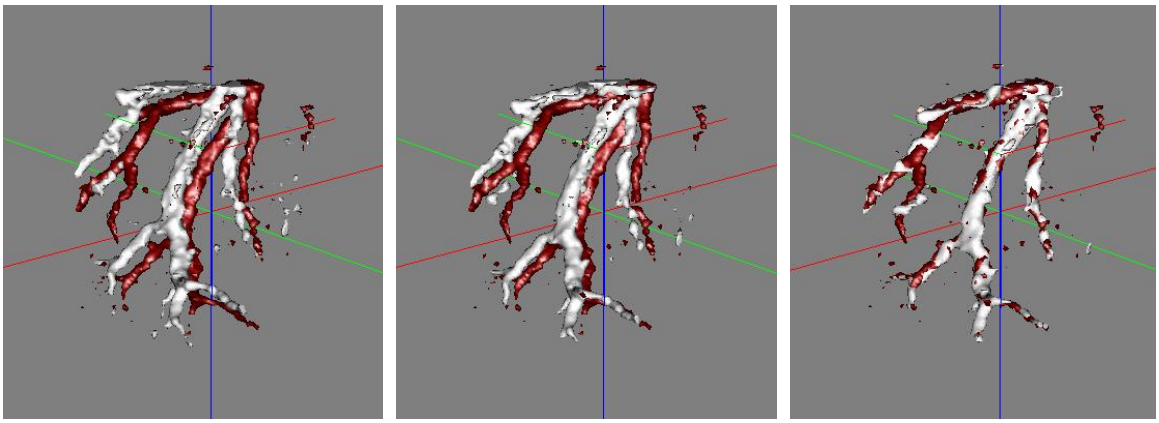


(d). The venous phase image (target image)

Figure 7. Axial images with manually-drawn vessel centerlines

Figure 8 shows representative results in 3D. The red vessels in Figure 8(a), (b) and (c) are identical. These are sub-trees of the portal vein in the venous phase image. The white vessels are the corresponding portal veins in the arterial phase image, and the rigidly and non-rigidly registered images, respectively. Results show that rigid registration alone cannot match the portal veins in the two imaging phases perfectly. Only

after non-rigid registration which is mainly driven by aligning vascular structures, can we get satisfactory results.



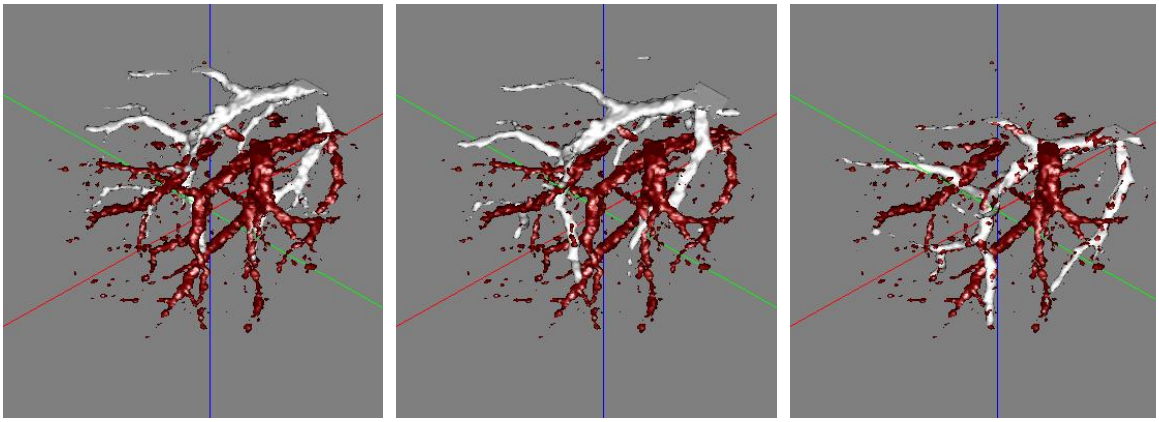
(a). The portal vein in the arterial phase image vs. that in the venous phase image

(b). The portal vein in the rigidly registered image vs. that in the venous phase image

(c). The portal vein in the non-rigidly registered image vs. that in the venous phase image

Figure 8. 3D registration results

Figure 9 is another example of results shown in 3D. In this example, besides the portal vein, the hepatic vein also appears in the venous phase image (also shown in red). Figure 9(c) shows that although the target image contains two different vessel systems, the non-rigid registration scheme does not produce any mismatches between different vessel systems.



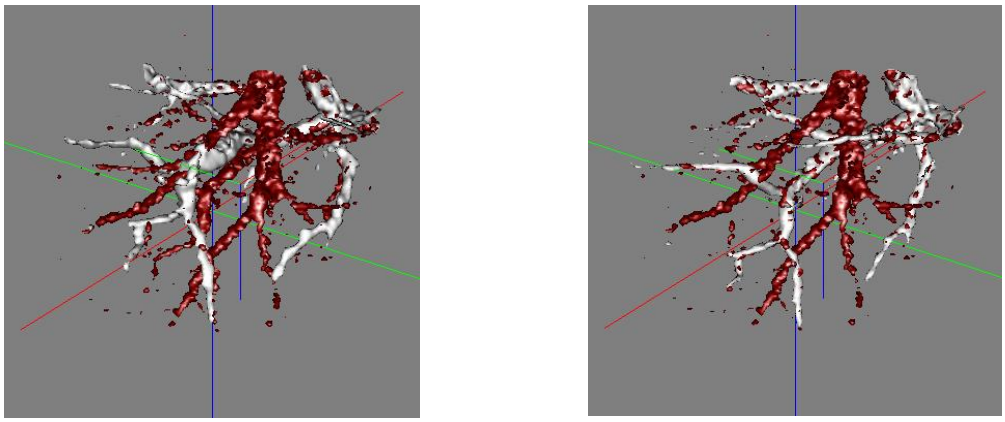
(a). The portal vein in the arterial phase image vs. that in the venous phase image

(b). The portal vein in the rigidly registered image vs. that in the venous phase image

(c). The portal vein in the non-rigidly registered image vs. that in the venous phase image

Figure 9. 3D registration results

Figure 10 shows the non-rigid registration results with and without intensity restriction. When the default intensity range is used (the minimum and maximum intensity of the image), the cost function (the normalized mutual information between the source and target images) is computed and maximized over the whole image, which includes the liver, the vessels and other surrounding structures. Since the vessels only occupy a small portion of the image foreground, the registration matches relatively large structures in the images well, but cannot guarantee perfect match between thin vessels (Figure 10(a)). When the intensity range only includes the range of intensities for vessels structures (1000-1250 in this example), the cost function is optimized within the vessels, and vessels are aligned very well (Figure 10(b)).



(a). Without intensity restriction (b). With intensity restriction between1000 and 1250

Figure 10. 3D non-rigid registration results

II.3 Hessian Enhancement Filter

Because the images we work with have low contrast to noise ratio, the first thing we do after registration is to denoise the image. Our goal is to preserve and enhance the vascular structures while smoothing the liver parenchyma. In [66] a vessel enhancement filter is proposed, which is based on the analysis of the second order local structure of an image. All three eigenvalues of the Hessian are incorporated into a vesselness measure, which offers an intuitive and geometrical interpretation for tubular structure detection. This filter is applied at different scale and the width of the vessel is estimated on the basis of the scale that maximizes the filter response.

II.3.1 Method Overview of the Hessian Filter

The idea of using multiscale second order local structure (Hessian) of an image to develop a vessel enhancement filter has been proposed first by Frangi et al. [66] and was

inspired by the work of Lorenz [67], and Sato [68]. Later this approach has been used and extended by numerous authors (see for instance [22]-[24], [29][30] and others).

This filter is used to determine a probability-like measurement that defines the likelihood that a pixel or voxel in an image belongs to a vessel. This can be accomplished by eigenanalysis of the Hessian at each point in the image. The eigenvalues of the Hessian measure the intensity variation in the direction of their corresponding eigenvectors. The direction of the eigenvector associated with the minimum eigenvalue indicates the direction of the minimum intensity variation, which is along the vessel.

Let H_σ be the Hessian matrix at a given voxel x, $H_\sigma(x) = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}$,

where I_{ij} denotes the second order partial derivatives of the Gaussian filtered image with the kernel size σ. Let the eigenvalues be λ_i ($|\lambda_1| \le |\lambda_2| \le |\lambda_3|$). The following table [66] shows various possibilities of the eigenvalues for the detection of different structures.

Table 1. Possible eigenvalues of the Hessian and corresponding patterns. H=high, L=low, N=noisy, usually small, +/- indicate the sign of the eigenvalues.

2D		3D			pattern
λ_1	λ_2	λ_1	λ_2	λ_3	
N	N	N	N	N	Noisy, no preferred direction
		L	L	H-	Plate-like structure (bright)
		L	L	H+	Plate-like structure (dark)
L	**H-**	**L**	**H-**	**H-**	**Tubular structure (bright)**
L	H+	L	H+	H+	Tubular structure (dark)
H-	H-	H-	H-	H-	Blob-like structure (bright)
H+	H+	H+	H+	H+	Blob-like structure (dark)

The discriminant function, which is called “vesselness”, developed in [66] can be expressed as

$$\nu(x,\sigma)=\begin{cases}0 & \lambda_2 \geq 0, \lambda_3 \geq 0 \\ \left[1-\exp\left(-\frac{R_a^2}{2\alpha^2}\right)\right]\exp\left(-\frac{R_b^2}{2\beta^2}\right)\left[1-\exp\left(-\frac{S^2}{2\gamma^2}\right)\right] & \textit{otherwise}\end{cases} \tag{3}$$

where $R_a=\left|\frac{\lambda_2}{\lambda_3}\right|$, $R_b=\frac{|\lambda_1|}{\sqrt{|\lambda_2\lambda_3|}}$, and $S=\|H_\sigma\|_F=\sqrt{\sum_j \lambda_j^2}$.

The quantities R_a, R_b, and S are designed to punish cross-sectional asymmetry, blobness and low energy, respectively. The vesselness measure is analyzed at different scales. The maximum of the filter response is achieved at a scale that approximately matches the size of the vessel to detect.

II.3.2 Feasibility of the Hessian filter

The multiscale vessel enhancement filter works well for angiography images with bright vessels in a relatively homogeneous dark background, such as MRA. In our application, we have observed that the artifacts resulting from the fact that the liver has been segmented and is surrounded by zero intensity values are so significant that they obscure the response from vessels within the parenchyma. From Figure 11 one can clearly see that the vesselness is very high around the liver border and much lower inside the liver.

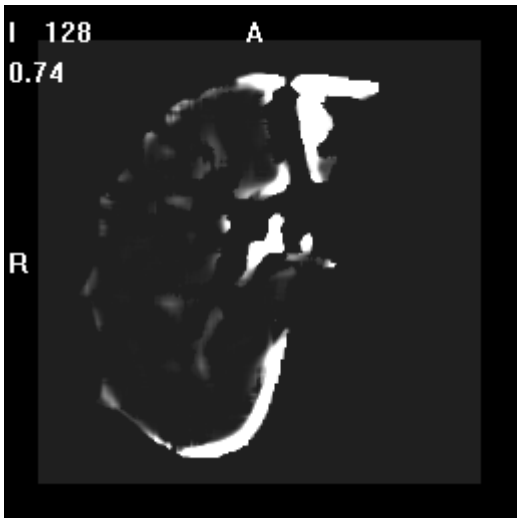


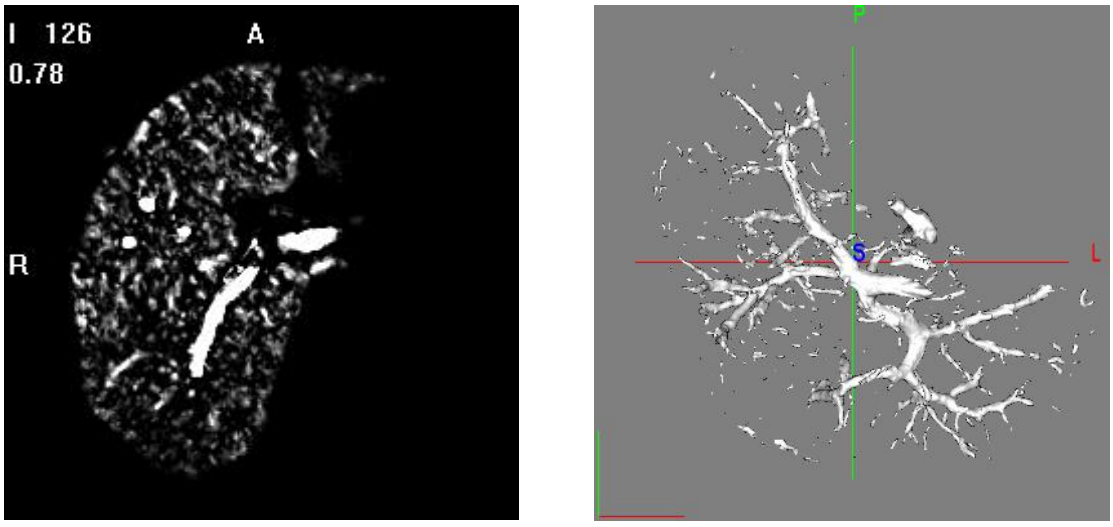
Figure 11. High vesselness results on liver boundary because of non-zero gradient

A simple way to alleviate this issue is to normalize the image first before Hessian filtering [52]. Here, the following transformation is applied to the image so that the intensity range of the image is mapped onto the [0, 1] interval[52]:

$$\tilde{I}(v) = \begin{cases} \dfrac{I(v) - g_{low}}{g_{high} - g_{low}} & if \quad g_{low} < I(v) < g_{high} \\ 1 & if \quad g_{high} \le I(v) \\ 0 & else \end{cases} \tag{4}$$

There are two parameters g_{low} and g_{high} used in this intensity remapping function. They are roughly the lower and upper bounds of the intensity values of the portal and hepatic veins in the abdominal CT images. Usually, the parameter g_{high} can be selected only once for images with similar intensity range. In our experience, the value of g_{low} needs to be adjusted for every volume, which is a major issue when trying to automate the process. To address this problem, we have developed an automatic method to estimate the best parameter value, which is described in Section II.3.3. The intensity

normalization reduces the big intensity difference between the liver tissue and the background, which, in turn, reduces the response of the vesselness filter around the borders of the liver. Figure 12(a) below shows the response of the hessian filter after normalization. For this data set, $g_{low} = 1140$, and $g_{high} = 1250$. Compared to the result without normalization (Figure 11), one can see that the vessels inside the liver are enhanced without producing much interference on the boundary. Figure 12(b) is the segmented vascular tree using just a threshold after Hessian enhancement. Although a lot of disconnected pieces are visible, this result is much clearer than the results obtained without the Hessian filter, which are shown in Figure 3(b).



(a). Hessian response after normalization (b). Thresholding result

Figure 12. A slice of the Hessian enhanced image and its thresholding result

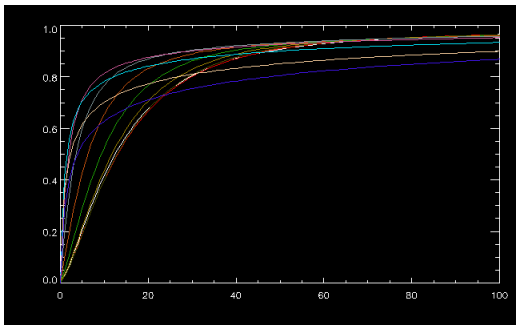
II.3.3 Automatic Selection of the Parameter g_{low} Used for Intensity Normalization

The automatic selection of g_{low} is based on the analysis of the cumulative probability function of the Hessian enhanced image and its derivative. To find the optimal g_{low} that produces the best enhancement, we apply the Hessian filter to the original image with several g_{low} values in a fixed range. In our experiments we have used 11 values between 1100 and 1200. For each of the Hessian enhanced images obtained with these values, we plot the cumulative intensity distributions as shown in Figure 13(a). All these curves have the same general shape: a sharp rise followed by a long plateau. The sharp rise corresponds to un-enhanced voxels (mainly liver parenchyma), the plateaus to voxels that have been enhanced by the filter. These include vessel voxels, border voxels, and spurious noisy voxels. We detect the plateaus by first computing the derivative of the cumulative density functions as shown in Figure 13(b). For each curve, we then select the beginning of the plateau as the intensity value at which the derivative is below a fixed threshold G_0 and we call this intensity value T_i. Here we have chosen $G_0 = 0.005$. Finally, we define the parameter N_i (percentage of high intensity voxels in the volume) for each g_{low},

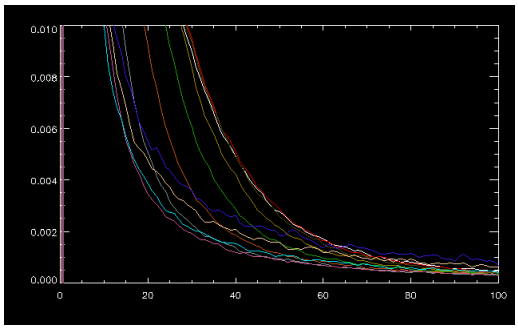
$$N_i = \frac{numnber\ of\ voxels\ with\ I > T_i}{number\ \ of\ \ liver\ \ voxels} \tag{5}$$

Figure 14(a) shows the value of N_i plotted versus g_{low}. We have observed that the optimal value of g_{low} is the value at which this curve has the largest negative slope, which makes intuitive sense. Indeed, with a small value for g_{low} a lot of liver parenchyma voxels are left in the image after normalization, so many border voxels are included in the

Hessian enhanced image. As g_{low} increases, liver tissue is gradually removed and fewer border voxels are included. When g_{low} reaches the range of values at which parenchyma and thus border voxels are eliminated, the count of high intensity voxels decreases rapidly. When all border voxels have disappeared, a further increase in the value of g_{low} only eliminates vessel voxels. Because there are fewer vessel voxels than liver voxels, the rate of change in the number of high intensity voxels decreases. To localize the optimal g_{low}, we thus fit a sixth order polynomial to the N_i's. We then compute its derivative and find its minimum. The derivative of the curve shown in Figure 14(a) is shown in Figure 14(b). The value at which the derivative is minimum is 1180, which is the optimal g_{low} value for this volume. Figure 15 shows enhanced images obtained with various values for g_{low} ranging from 1150 to 1200. This figure shows that the results obtained with the optimal value g_{low} preserves all the details of the vessel trees while eliminating undesired border voxels.

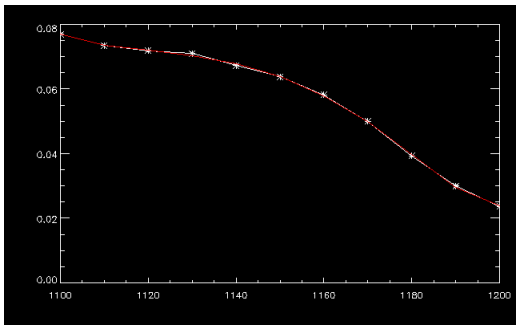


(a). The cumulative probability functions for 11 g_{low}

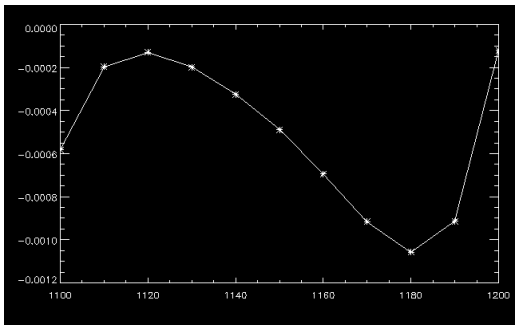


(b). The derivative of the cumulative probability functions

Figure 13. The cumulative probability functions and their derivatives

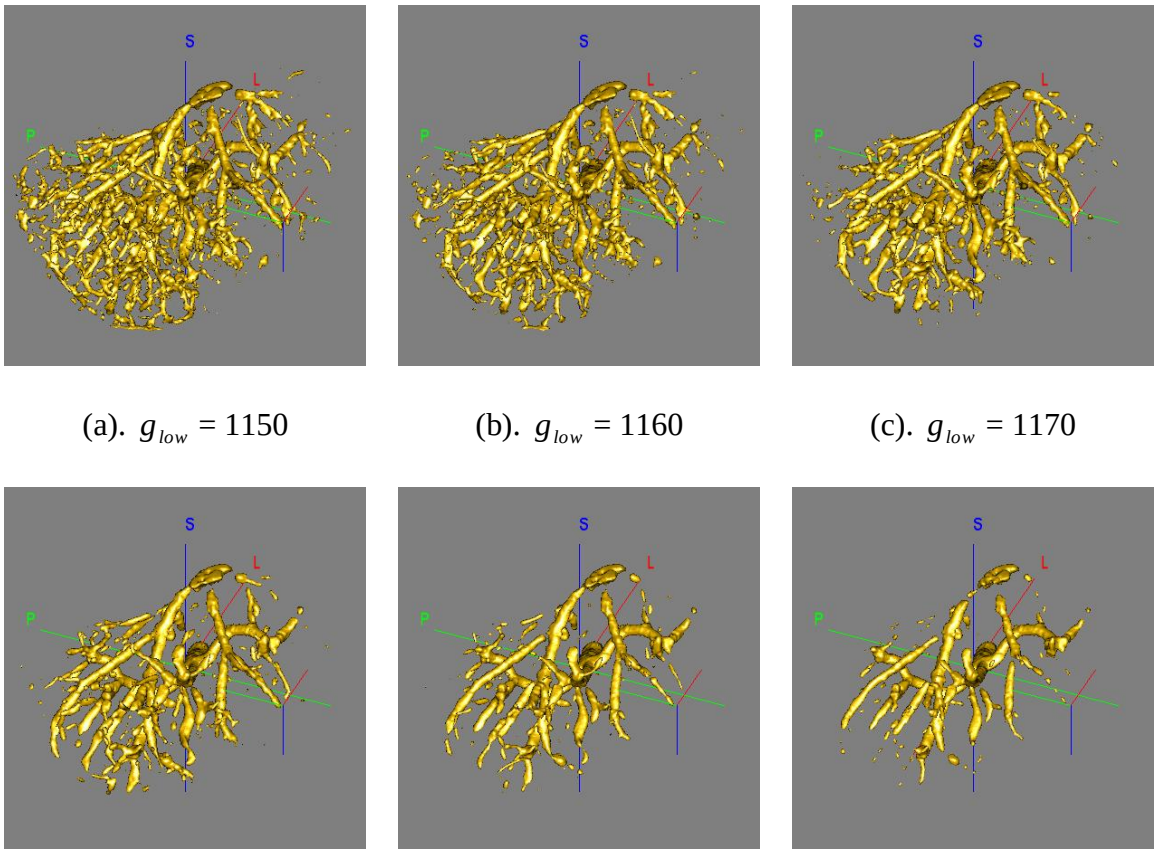


(a). N_i vs. g_{low}



(b). The derivative of N_i vs. g_{low}

Figure 14. The plot of N_i vs. g_{low} and its derivative



(a). g_{low} = 1150 (b). g_{low} = 1160 (c). g_{low} = 1170

(d). g_{low} = 1180 (e). g_{low} = 1190 (f). g_{low} = 1200

Figure 15. Comparison of the enhanced vessels with different g_{low} values

Computing Hessian enhancement at multiple scales is essential as a preprocessing step before vessel segmentation, since small scale improves the contrast for small vessels, while large scale highlights vessels with large radius. However, to obtain the optimal g_{low} value for each data volume, we need to apply the filter several times, and multiple scale computing is time-consuming. To reduce the computation time required to obtain the optimal g_{low}, the Hessian filter can be applied to the image with a single medium scale. We use a scale = 3 voxels for all 5 data sets and the resulting optimal g_{low} are the same as those obtained with multiple scales enhancement. The automatic method we have

developed for the selection of g_{low} led to satisfactory results for all the volumes included in our study.

II.3.4 Parameters Selection for the Vesselness Equation

There are 3 parameters α, β and γ that need to be set in the vesselness equation, which for convenience, is rewriten below:

$$\nu(x,\sigma)=\begin{cases}0 & \lambda_2 \geq 0, \lambda_3 \geq 0\\ \left[1-\exp\left(-\frac{R_a^2}{2\alpha^2}\right)\right]\exp\left(-\frac{R_b^2}{2\beta^2}\right)\left[1-\exp\left(-\frac{S^2}{2\gamma^2}\right)\right] & \textit{otherwise}\end{cases}$$

where $R_a=\left|\frac{\lambda_2}{\lambda_3}\right|$, $R_b=\frac{|\lambda_1|}{\sqrt{|\lambda_2\lambda_3|}}$, and $S=\|H_\sigma\|_F=\sqrt{\sum_j \lambda_j^2}$

The ratio R_a distinguishes plate-like and tubular structures with a value close to 0 for the former and close to 1 for the latter. The ratio R_b accounts for blob-like structures. It tends to 1 for a blob-like structure but tends to 0 for the other two types of structures. The 3 parameters α, β and γ control the sensitivity of the tubular detection filter by assigning different weights to R_a, R_b, and S.

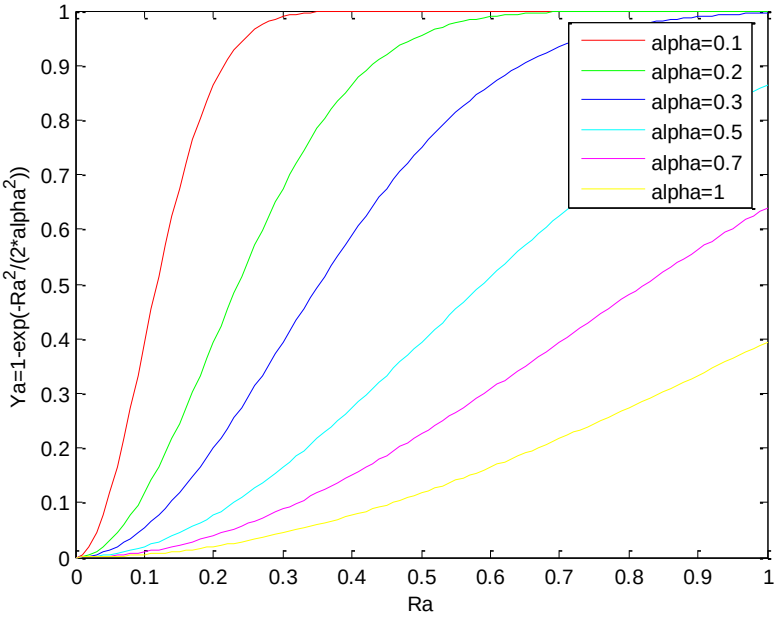


Figure 16. Y_a vs. R_a plot

Figure 16 shows the relationship between R_a and $Y_a = 1 - \exp\left(-\frac{R_a^2}{2\alpha^2}\right)$ as α changes from 0 to 1. When α is selected small, for example, $\alpha = 0.1$, Y_a is very large (close to 1) and does not change much as R_a changes from 0.3 to 1, which means there is no big difference in the vesselness value between plate-like structures (R_a tends to 0.3) and tubular structures (R_a tends to 1). As a consequence, with low values of α the filter will not discriminate between plate-like and tubular structures. This can be seen experimentally. With low values of α there are too many plate-like structures left in the filtered image. Figure 16 above shows that α should be at least equal or greater than 0.3 to maximize the discrimination power of the filter between plate-like and tubular

structures. If α is chosen too high, e.g. $\alpha = 0.7$, vessels will become disconnected because at their junctions they look more like plates than tubes.

The relationship between R_b and $Y_b = \exp\left(-\frac{R_b^2}{2\beta^2}\right)$ as β changes from 0 to 1 is similar to what is shown in Figure 16. Small β values cause Y_b to decreases to 0 rapidly and it is not easy to distinguish the two cases, $R_b \to 0$ and $R_b \to 1$. The best approach is to choose a value for β so that when R_b changes from 0 to 1, Y_b varies between 0 and 1 gradually, so Y_b, and thus the total filter response ν, can be used to distinguish between large R_b (blob-like structures) and small R_b (the other two cases). So, as is the case for α, β should be at least equal or greater than 0.3. Experimental results show consistently more noise and short disconnected lines for $\beta = 0.1$ than for greater β values. Better results are obtained with larger β values.

The parameter γ is set as a certain percentage of the maximum intensity value of the image. Low γ produces too much noise and makes the small vessels disappear. To sum up, through analysis and experiments, we have set the parameter values to $\alpha = 0.3, \beta = 0.7, \gamma = 0.5 * I_{\max}$. These values produced good results for all our data sets.

CHAPTER III

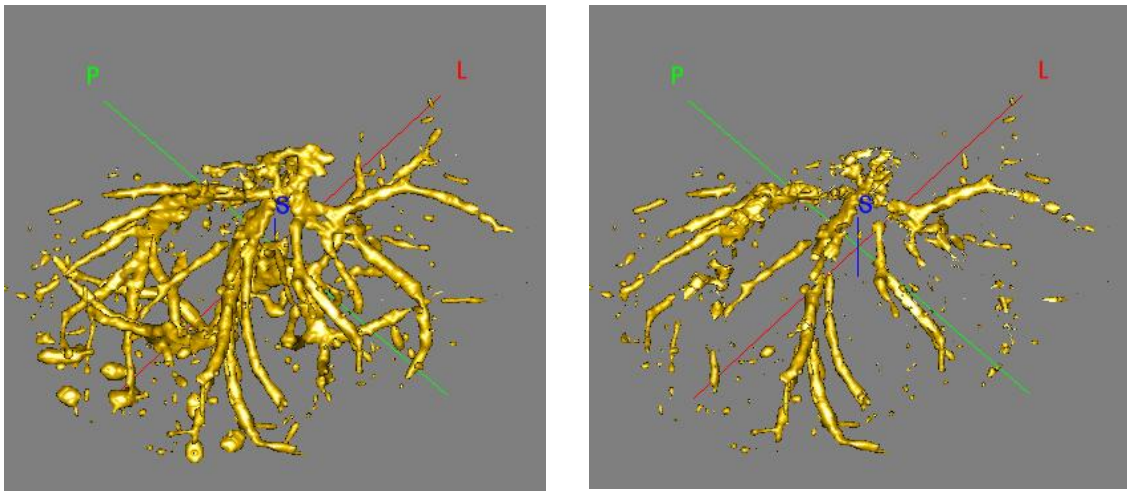
HEPATIC VESSEL SEPARATION

Due to the scanning protocol used to generate the images used in this work, both the arterial and the venous phase images contain the portal vein, and the venous phase images also contain the hepatic vein. These two types of veins are connected to each other at a number of points due to the resolution of the images. It is thus very difficult to extract them separately. To obtain accurate segmentation of both veins, the first step is to separate these two vascular systems. Because of the characteristic of the two imaging phases, one simple way to separate the two trees could be to subtract the portal vein of the arterial phase from the venous phase, which could result in an image that contains only the hepatic vein. Such a method is intuitive and fast, but lacks accuracy and robustness. It may lead to disconnected pieces of the hepatic vein and unwanted noise in the resulting image. We have developed an alternative method for separation, which is based on 2D histogram analysis. We got this idea by noticing that the portal vein and the hepatic vein pixels form two separable clusters in the 2D intensity histogram, which can be used for classification of the different vessel pixels. In later subsections of this chapter, we will describe both separation methods (ie., separation obtained via subtraction and separation obtained with the new method we have developed) in more details and then compare their results.

III.1 Vessel Separation by Subtraction

After registration and Hessian enhancement, the portal vein in an arterial phase image is aligned with the portal vein in the corresponding venous phase image, which also contains the hepatic vein. In principle, subtracting the portal vein from the venous phase image could result in an image that contains the hepatic vein only. But, simple subtraction of the gray level images does not work very well because of contrast differences between the two phases. Rather, we first binarize the arterial phase image and then subtract it from the venous phase image. To do this a threshold T_PV needs to be applied to the Hessian enhanced arterial phase image to identify candidate portal vein voxels. Then these portal vein pixels are removed from the Hessian enhanced venous phase image. With another threshold T_HV applied to this image, hepatic vein voxels are labelled. We have observed two important things when experimenting with real images. First, the final results are very sensitive to the selection of the two intensity thresholds and their values depend on the images. Second, even if the thresholds are selected manually to lead to the best possible results, a simple subtraction of the binarized images does not lead to optimal results.

The figures below show venous phase images before and after vessel separation using vessel subtraction. The image before separation contains the portal vein and the hepatic vein. After processing, the venous phase image includes the hepatic vein only.



(a). Venous phase image before separation (b). Venous phase image after separation

Figure 17. Venous phase image before and after separation

As shown in this example, simple subtraction between the two image phases leads to disconnected vessel branches and spurious noise pixels. To both address the intensity threshold selection problem and improve on vessel separation once the thresholds are selected, we have developed a method, which is based on characteristics of the 2D intensity histogram computed with the registered arterial and venous phase images.

III.2 Vessel Separation by Histogram Analysis

In the following subsections, phrases such as the arterial/venous phase image and the portal/hepatic vein, will appear frequently. For simplicity and convenience, we use the abbreviations instead of the full names. That is, we use AP/VP for the arterial/venous phase image and PV/HV for the portal/hepatic vein.

III.2.1 Method Overview

The arterial phase (AP) images contain the portal veins (PV), and the venous phase (VP) images contain both the portal veins and the hepatic veins (HV). Figure 18 is an example of the 2D histogram of the AP and VP images (intensities from the AP are plotted along the horizontal axis and those from the VP on the vertical axis), where one cannot even tell vascular structures from liver parenchyma, due to the fact that the volume of the vessels is very small compared to the whole liver volume.

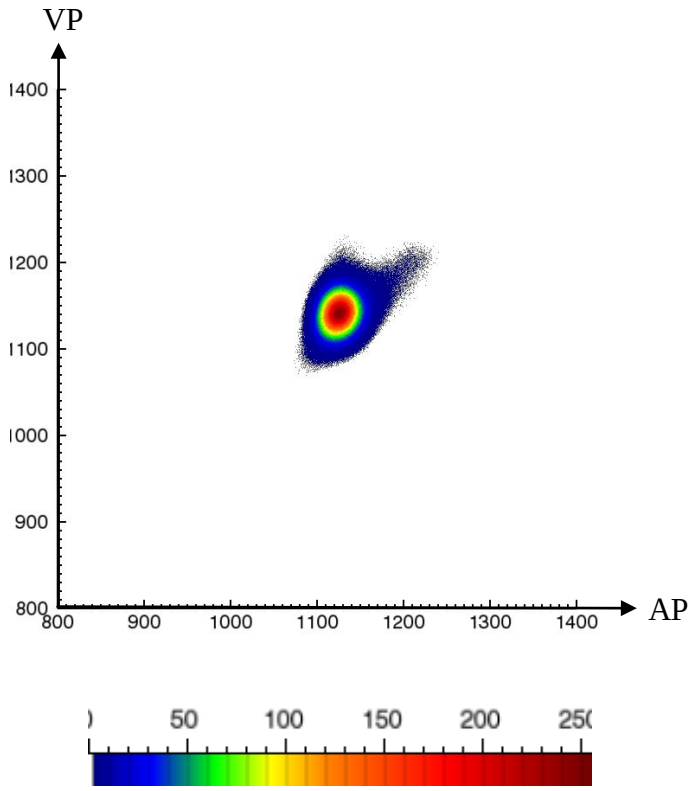
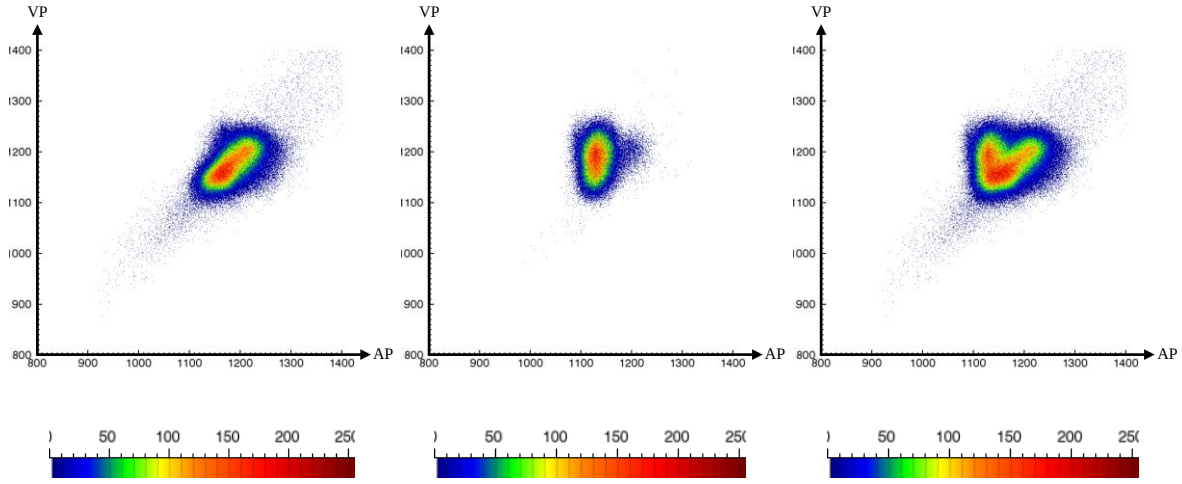


Figure 18. An example of the 2D histogram of the AP and VP images

Clearly, not enough information can be obtained from the 2D histogram of the AP and VP images for vessel separation. However, when we restrict the 2D histogram to the

pixels of the two vessels, the PV and HV, obtained using the separation method described in Section III.1, things are different.

In III.1, the PV mask is obtained from the Hessian enhanced AP image with a threshold T_PV. Then these PV pixels are removed from the Hessian enhanced VP image, and an image, which contains mostly the HV, is produced. With another threshold T_HV, the HV mask is computed. 2D histograms for PV and HV voxels can thus be generated, and they are shown in Figure 19.



(a). The PV histogram (b). The HV histogram (c). The PV+HV histogram

Figure 19. An example of the 2D PV and HV histograms

These figures show that segmented PV pixels have high intensities in both phases, so the PV pixels are mainly located in the top right of the histogram; segmented HV pixels have high intensities in the VP image but low intensities in the AP, so the HV pixel cluster mainly located in the top left of the histogram. In practice, after normalization and summation, these two pixel clusters overlap with each other to a certain degree because of imaging limitations and non-perfect registration, but they still appear separable (Figure

19.). As will be discussed later a statistical classifier is used to separate the two clusters to segment the vessels and results will show that this produces better results than simple subtraction. But parameter estimation for this classifier requires knowing a priori which pixel belongs to which class, i.e., vessel segmentation. To address this issue, we initiate the process by selecting the thresholds T_PV and T_HV (the exact procedure used to do this will be detailed in Section III.2.2) and we use these thresholds to generate the initial PV and HV masks. These masks are used to estimate the initial classifier parameters. Once the classifier is estimated, it is used to separate the entire image into two classes (i.e., PV or HV) and pixels in each of these classes with a high value in the corresponding Hessian image are labeled as either PV pixels or HV pixels. Using these pixels, the classifier's parameters are re-estimated and the process is repeated until convergence. In our experience, this approach works well if good values for the two initial thresholds can be found. In Section III.2.2 we describe a method we have developed to automate the estimation of these parameters. Figure 20 illustrates the complete procedure we have developed.

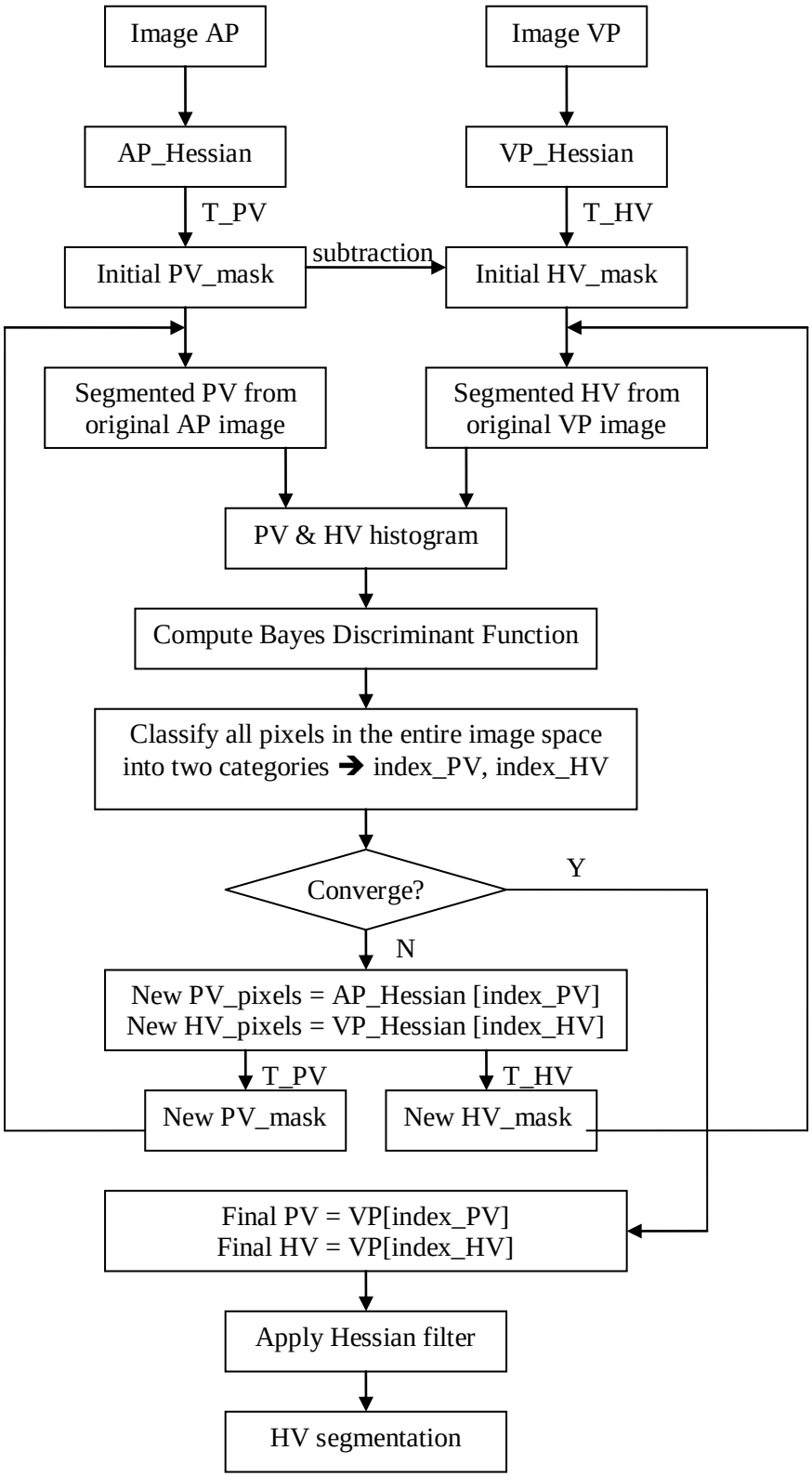
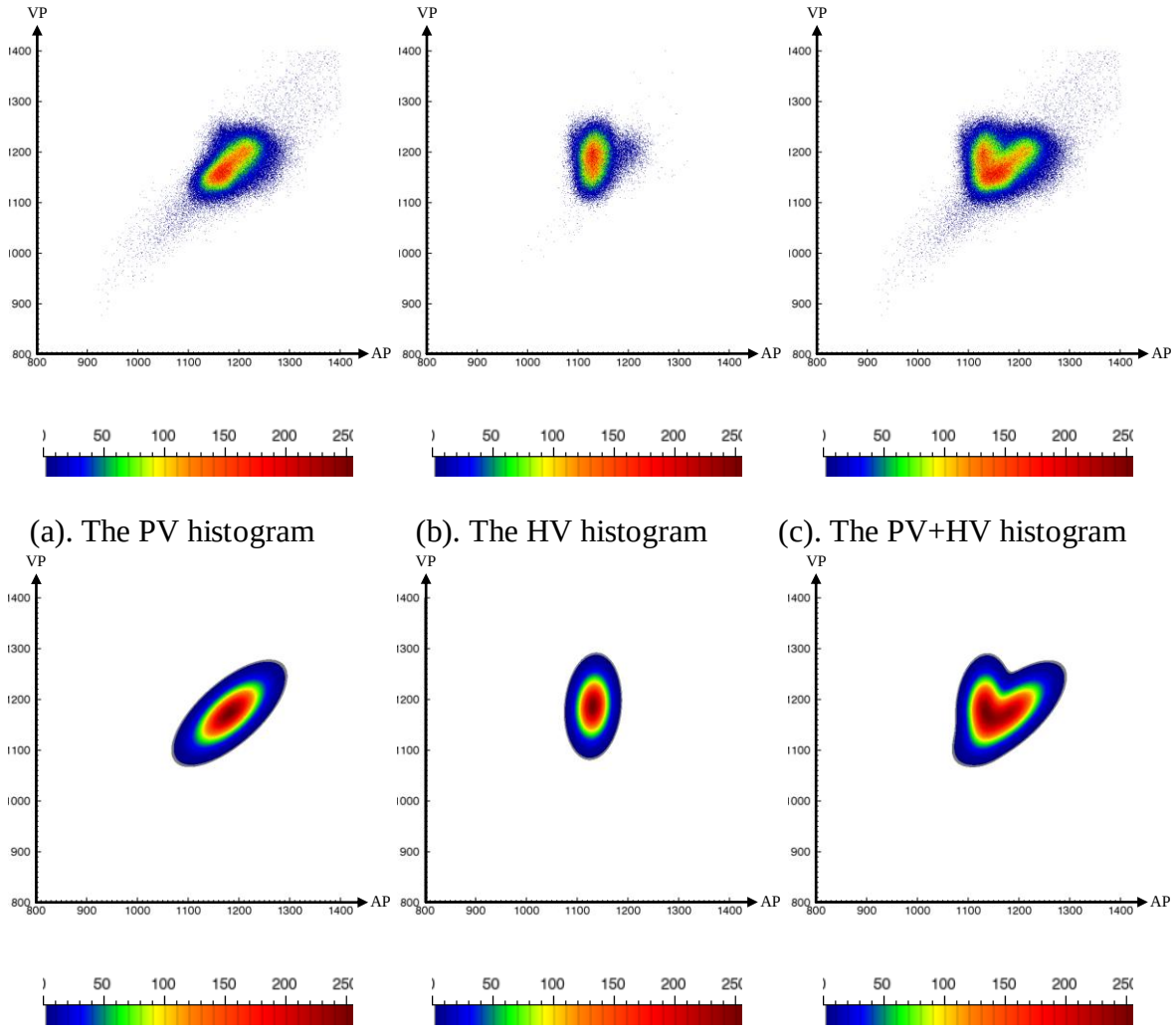


Figure 20. Flowchart of the process of separation by histogram analysis

To classify the pixels into two classes, we have used a standard Bayesian classifier. The shape of the 2D histograms both for the PV and HV pixels suggests that the distributions can be approximated by Gaussians, which we have used. Figure 21 shows the 2D histograms (top panels) for the PV pixels, the HV pixels, and both classes. The bottom panels show the isocontours for the estimated Gaussian distributions.



(a). The PV histogram (b). The HV histogram (c). The PV+HV histogram

(d). Gaussian fitting of the PV histogram (e). Gaussian fitting of the HV histogram (f). Gaussian fitting of the PV+HV histogram

Figure 21. Comparison of PV, HV histograms with their Gaussian fittings

A 2D Gaussian probability density function is written as:

$$p(\vec{x}) = \frac{1}{2\pi|\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(\vec{x}-\vec{\mu})^T \Sigma^{-1}(\vec{x}-\vec{\mu})\right] \quad (6)$$

where $\vec{x}$ is a 2-component column vector, $\vec{\mu}$ is the 2-component mean vector, $\sum$ is the 2-by-2 covariance matrix, and $|\sum|$ and Σ^{-1} are its determinant and inverse, respectively. And $(\vec{x}-\vec{\mu})^T$ denotes the transpose of $(\vec{x}-\vec{\mu})$.

According to Bayes decision theory, the minimum-error-rate classification can be achieved using the discriminant function

$$g(\vec{x}) = \ln \frac{p(\vec{x}\mid c_1)}{p(\vec{x}\mid c_2)} + \ln \frac{P(c_1)}{P(c_2)} \quad (7)$$

A pixel is assigned to class c_1 if $g(\vec{x}) > 0$, otherwise it is assigned to class c_2. $P(c_i)$ is the prior probability of each class, and $p(\vec{x}\mid c_i)$ is the conditional probability. Assuming equal prior probabilities and normal density function, that is, $P(c_1) = P(c_2) = \frac{1}{2}$ and $p(\vec{x}\mid c_i) \sim N(\vec{\mu}_i, \Sigma_i)$, after simplification, the discriminant function can be rewritten as:

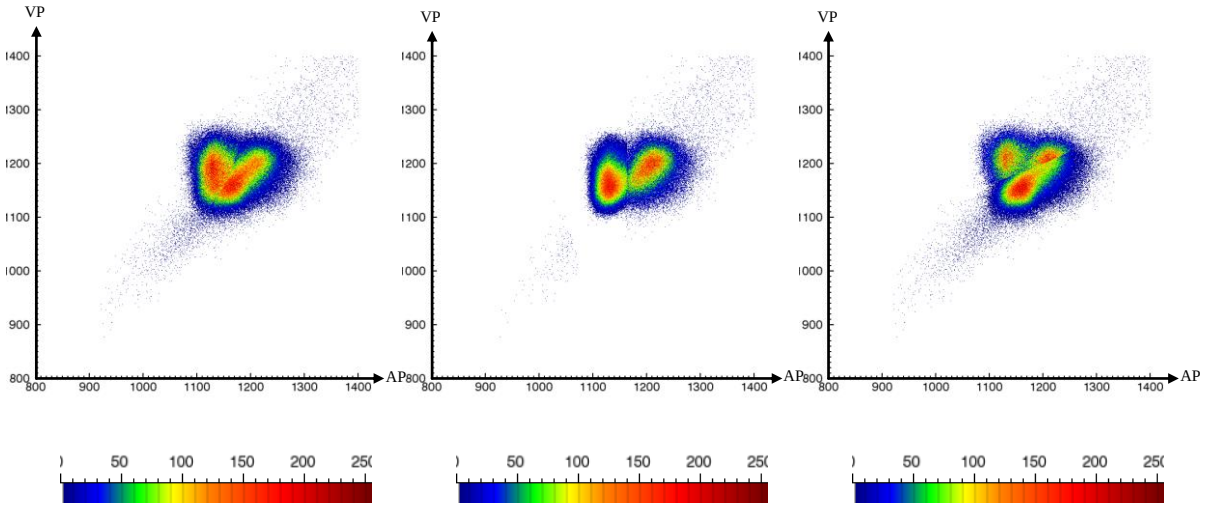
$$g(\vec{x}) = -\frac{1}{2}\left[(\vec{x}-\vec{\mu}_1)^T \Sigma_1^{-1}(\vec{x}-\vec{\mu}_1) - (\vec{x}-\vec{\mu}_2)^T \Sigma_2^{-1}(\vec{x}-\vec{\mu}_2)\right] - \frac{1}{2}\ln\frac{|\Sigma_1|}{|\Sigma_2|} \quad (8)$$

As discussed above, all pixels in both AP and VP images are classified into either the PV or the HV class, depending on the value of the discriminant function, and the process iterates until the number of pixels in each class does not change.

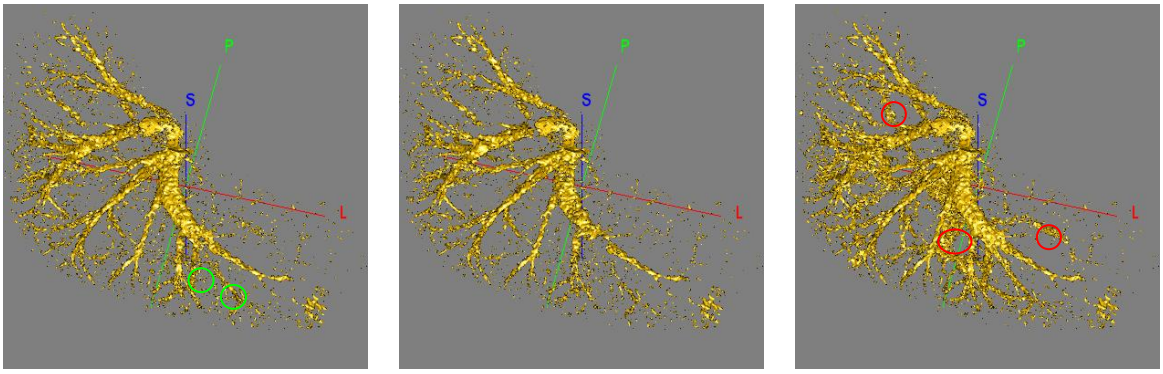
III.2.2 Automatic Threshold Selection

As stated in the previous section, our algorithm requires two thresholds: T_PV and T_HV. In this section, we discuss how to select these two parameters automatically. First, we observe that when these two thresholds are changed, a different numbers of vessel pixels are recruited into the masks of the PV and HV, in the first and subsequent iterations, which affect the separation results.

Figure 22 shows the 2D histograms and the separation results obtained with three pairs of thresholds. In each case, the thresholds were chosen manually and vessel separation was achieved using the classification method described in the previous section. The threshold pair resulting in the 2D histogram shown in Figure 22 (a) results in the best vessel separation. The threshold pair resulting in the 2D histogram shown in Figure 22 (b) leads to results in which HV voxels are missed (green circles). Figure 22 (c) is another example of suboptimal thresholds. The separation results shown in Figure 22 (f) are almost useless because they contain a lot of PV voxels. Visual inspection of many 2D histograms obtained with various threshold pairs suggest that the shape of the 2D histogram correlates with separation results. As a rule, better separation results are obtained when the histograms for both classes are spread and when they are maximized in size.



(a). Optimal thresholds (b). Sub-optimal thresholds (c). Sub-optimal thresholds

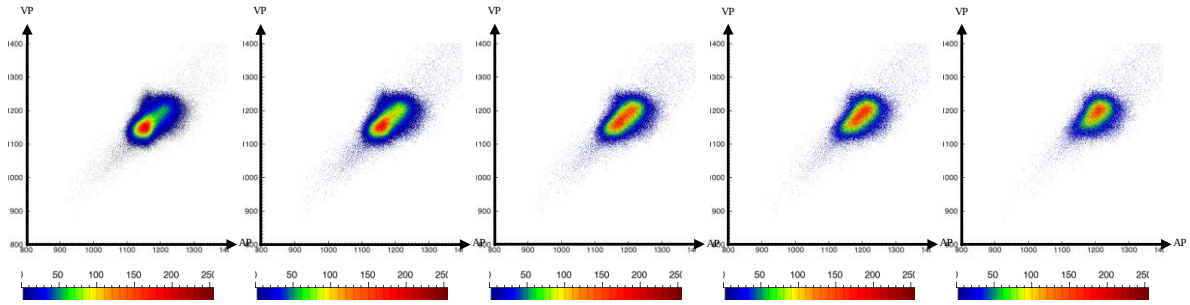


(d). Separated HV from (a) (e). Separated HV from (b) (f). Separated HV from (c)

Figure 22. Various separation results from different thresholds

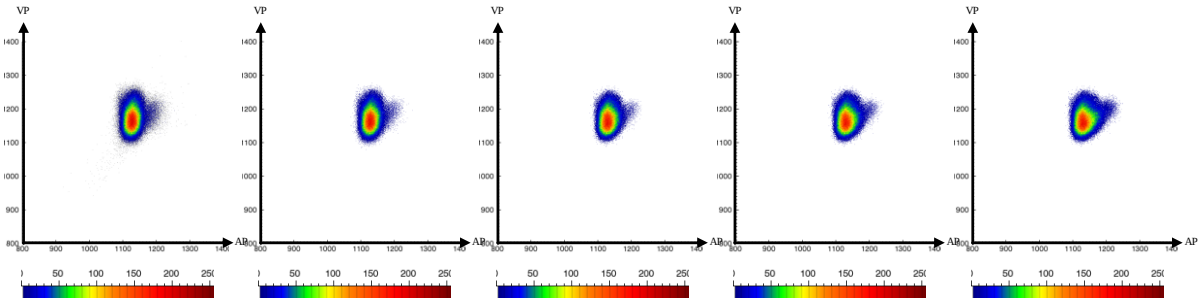
Visual observation of these histograms also permits us to derive certain rules. When T_PV increases and T_HV is fixed, the number of PV pixels decreases, so the PV pixel cluster gets smaller and the center of gravity of the cluster moves higher (Figure 23). The HV pixels do not change much with lower T_PV values, but the cluster gets more spread as PV pixels get recruited (Figure 24). This can be explained as follows: when T_PV is set high, more pixels, including suspicious PV pixels, remain in the HV mask after the subtraction of the PV pixels. On the other hand, if we fix T_PV while increasing

T_HV, the PV pixel cluster does not change (Figure 25), but the HV pixel cluster becomes smaller and its center of gravity moves higher (Figure 26). The following figures show the shape of the histograms as one of the thresholds is changed.



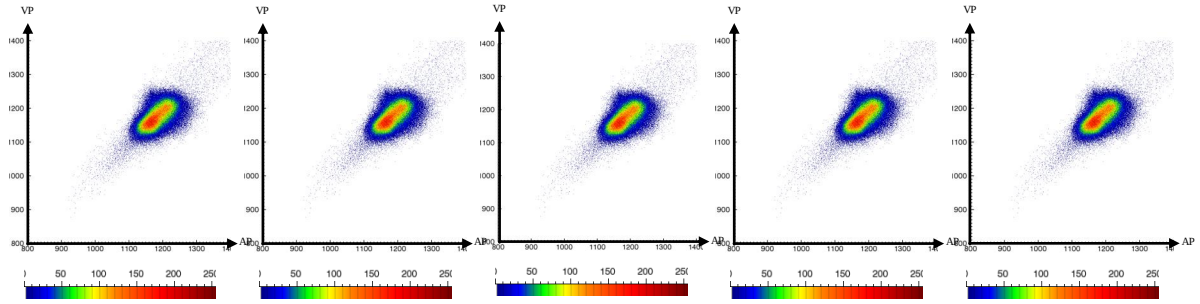
T_PV = 10 T_PV = 20 T_PV = 40 T_PV = 60 T_PV = 100

Figure 23. The PV histogram, as T_PV changes while T_HV is fixed (T_HV=10)



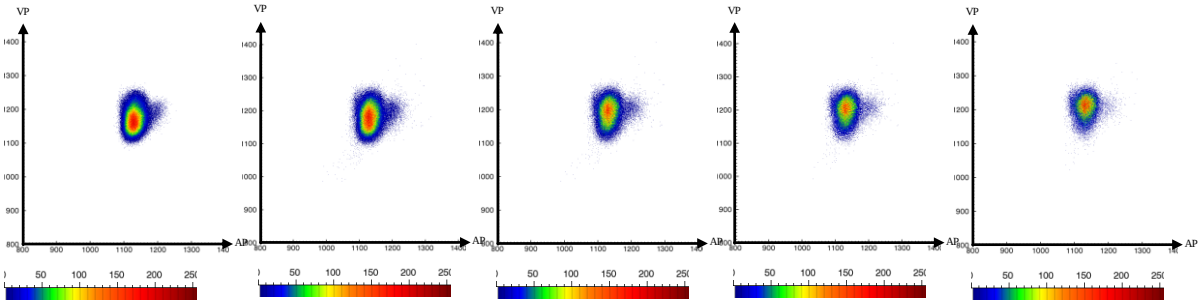
T_PV = 10 T_PV = 20 T_PV = 40 T_PV = 60 T_PV = 100

Figure 24. The HV histogram, as T_PV changes while T_HV is fixed (T_HV=10)



T_HV=10 T_HV=20 T_HV=40 T_HV=60 T_HV=100

Figure 25. The PV histogram, as T_HV changes while T_PV is fixed (T_PV=30)



T_HV=10 T_HV=20 T_HV=40 T_HV=60 T_HV=100

Figure 26. The HV histogram, as T_HV changes while T_PV is fixed (T_PV=30)

In our experience when both PV and HV histogram sizes are maximized and evenly distributed, optimal separation is achieved (Figure 22). Finding optimal thresholds thus become a matter of finding a quantity that characterizes the size and shape of the histograms.

Shannon entropy is widely used in information theory. It is a measure of the uncertainty associated with a random variable, whose probabilities are given by $p(s)$:

$$H = -\sum_{s} p(s) \log p(s) \tag{9}$$

The entropy is maximum when all the outcomes are equiprobable, i.e., when the probability distribution is uniform. The entropy is minimum when the distribution is an impulse. The spread of the joint histogram is thus related to the joint entropy: the more the joint histogram is spread, the higher the entropy. Following our observations, we need to find the thresholds T_PV and T_HV that maximize the joint entropy:

$$H = -\sum_{j,k} PDF[j,k] \log PDF[j,k] \tag{10}$$

where j and k are intensities from the AP and VP images. To obtain the joint probability distribution function (PDF) we normalize the joint histogram:

$$PDF[j,k] = \frac{Histogram[j,k]}{\sum_{j,k} Histogram[j,k]} \tag{11}$$

But using the joint entropy alone is not enough to find the optimal threshold for the HV histogram. This is so because as T_PV increases, PV pixels appear in the HV histogram. This causes the HV histogram to disperse and the entropy to increase (Figure 24). This needs to be prevented. Looking at Figure 22, we notice that a proper threshold produces a Gaussian-shaped histogram, with a long axis and a short axis orthogonal to each other; when PV pixels start appearing in the HV histogram, the length of the short axis, as well as the ratio between the short and long axis increases. The two eigenvectors of the covariance matrix are the two principal components of the 2D histogram. The eigenvalues give the variance of the histogram along the two principal component directions. So the ratio between the two eigenvalues captures the shape of the histogram.

The smaller the ratio is, the more elongated the shape of the histogram is, and the better the final separation is.

We thus combine the entropy measurement and the ratio of the eigenvalues to automatically select the thresholds that maximize M (this is done separately for the HV and PV histograms).

$$M = Entropy - \frac{SmallEigenvalue}{Largeeigenvalue} \tag{12}$$

As we vary both the T_PV and T_HV thresholds, the measurement M is computed for each combination of the two thresholds. Table 2 and 3 show the value of the measurement M for the PV and HV histogram, respectively, as T_PV and T_HV change from 10 to 100. Because the PV histogram depends only on T_PV and does not change when T_HV is changed, the value of M for the PV varies only along the column (as shown in the green box in Table 2) and does not change along the rows of Table 2. We thus select the optimal T_PV by maximizing M, as shown in the red ellipse. Then we fix T_PV at this optimal value (T_PV=40 in this example, as shown in the red frame in Table 3). Table 3 shows the effect of varying T_HV for fixed values of T_PV. The optimal value for T_HV is chosen as the value that maximizes M for the HV histogram with the previously selected optimal T_PV value. (T_HV=20 in this example, shown in the blue ellipse). With the selected T_PV and T_HV, we follow the separation steps described in III.2.1.

Table 2. Measurement M of the PV histogram

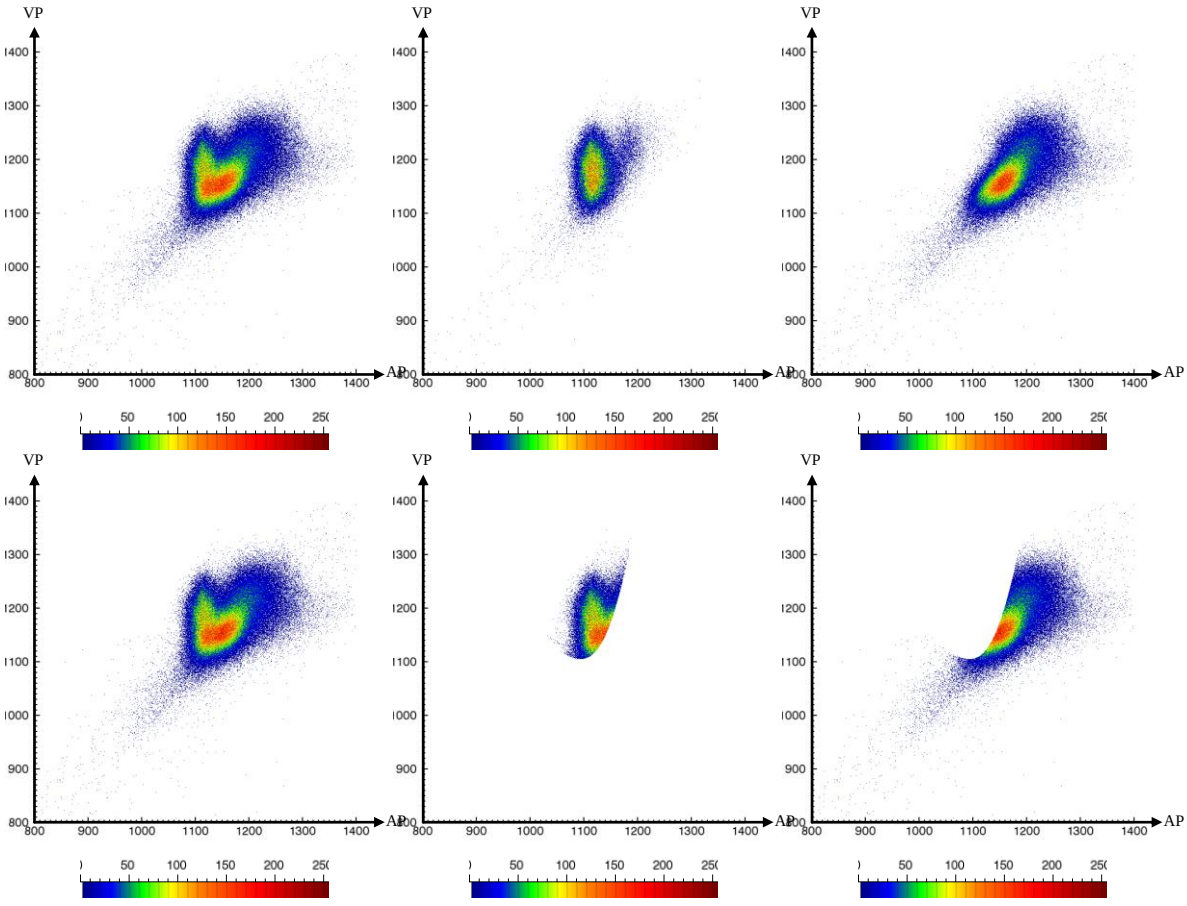
PV	T_HV=10	T_HV=20	T_HV=30	T_HV=40	T_HV=50	T_HV=60	T_HV=70	T_HV=80	T_HV=90	T_HV=100
T_PV=10	3.81580	3.81580	3.81580	3.81580	3.81580	3.81580	3.81580	3.81580	3.81580	3.81580
T_PV=20	3.82388	3.82388	3.82388	3.82388	3.82388	3.82388	3.82388	3.82388	3.82388	3.82388
T_PV=30	3.82543	3.82543	3.82543	3.82543	3.82543	3.82543	3.82543	3.82543	3.82543	3.82543
T_PV=40	3.82672	3.82672	3.82672	3.82672	3.82672	3.82672	3.82672	3.82672	3.82672	3.82672
T_PV=50	3.82341	3.82341	3.82341	3.82341	3.82341	3.82341	3.82341	3.82341	3.82341	3.82341
T_PV=60	3.81482	3.81482	3.81482	3.81482	3.81482	3.81482	3.81482	3.81482	3.81482	3.81482
T_PV=70	3.80524	3.80524	3.80524	3.80524	3.80524	3.80524	3.80524	3.80524	3.80524	3.80524
T_PV=80	3.79427	3.79427	3.79427	3.79427	3.79427	3.79427	3.79427	3.79427	3.79427	3.79427
T_PV=90	3.78002	3.78002	3.78002	3.78002	3.78002	3.78002	3.78002	3.78002	3.78002	3.78002
T_PV=100	3.76582	3.76582	3.76582	3.76582	3.76582	3.76582	3.76582	3.76582	3.76582	3.76582

Table 3. Measurement M of the HV histogram

HV	T_HV=10	T_HV=20	T_HV=30	T_HV=40	T_HV=50	T_HV=60	T_HV=70	T_HV=80	T_HV=90	T_HV=100
T_PV=10	3.66740	3.60694	3.53859	3.46128	3.40380	3.34079	3.27625	3.21657	3.14214	3.07956
T_PV=20	3.69198	3.64052	3.58037	3.49618	3.46360	3.40997	3.34900	3.28711	3.23619	3.18108
T_PV=30	3.62382	3.62513	3.56085	3.52623	3.46646	3.40835	3.35140	3.31959	3.28775	3.24001
T_PV=40	3.64553	3.66730	3.57481	3.51408	3.44556	3.44843	3.38364	3.32484	3.31679	3.26475
T_PV=50	3.64823	3.55224	3.60732	3.52045	3.45246	3.40459	3.34431	3.36530	3.30730	3.24954
T_PV=60	3.66341	3.55243	3.46307	3.55591	3.49776	3.43006	3.38419	3.31850	3.27824	3.30353
T_PV=70	3.61417	3.55840	3.46779	3.36489	3.50513	3.44337	3.39405	3.33958	3.29499	3.23485
T_PV=80	3.61657	3.56149	3.46673	3.36787	3.50932	3.44556	3.39446	3.33828	3.30075	3.23333
T_PV=90	3.61773	3.56610	3.46864	3.37535	3.27717	3.27717	3.40589	3.34488	3.30583	3.24770
T_PV=100	3.61696	3.56559	3.46770	3.36955	3.27020	3.46878	3.41271	3.35044	3.30654	3.24643

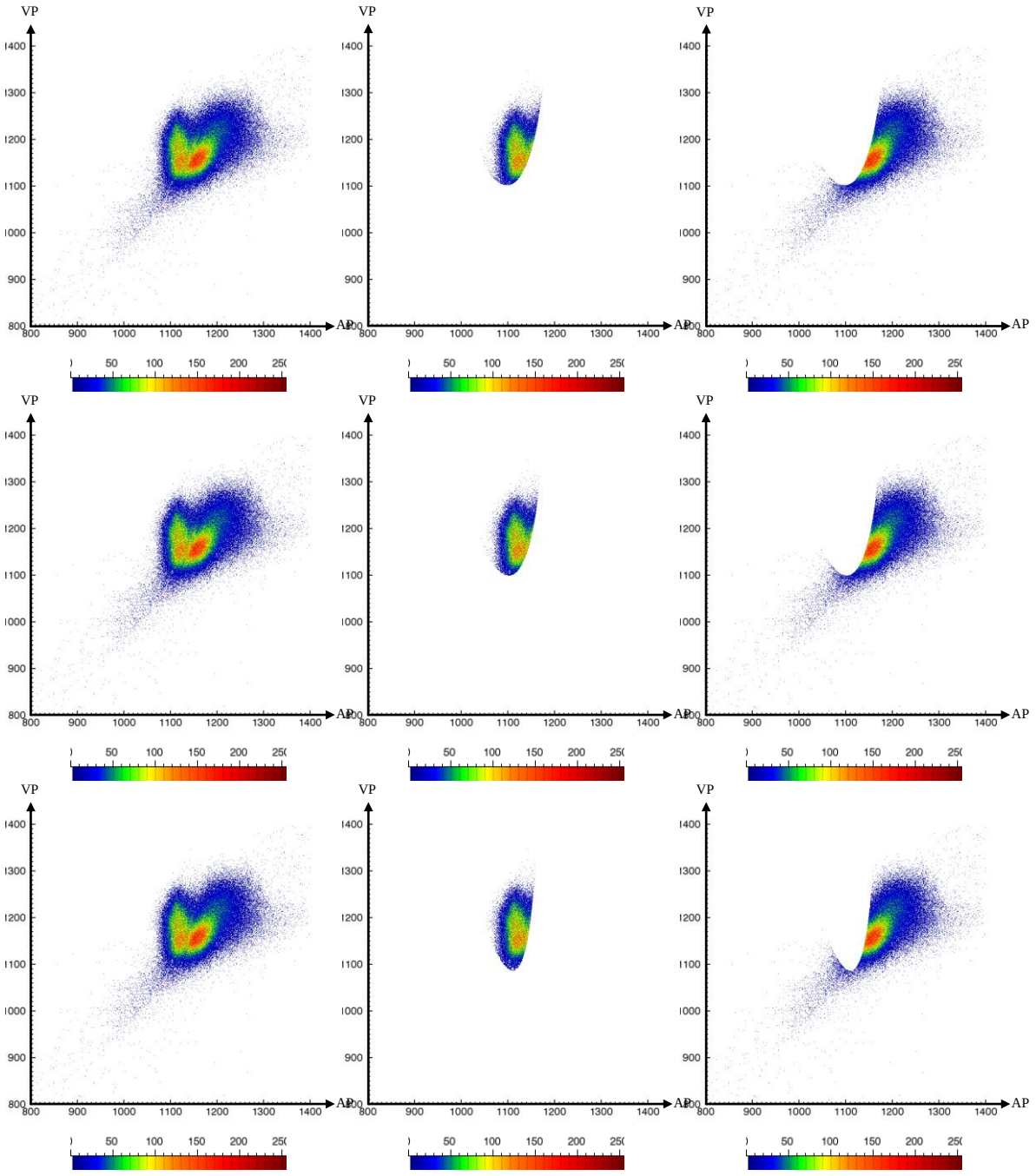
III.3 Results and Discussion

Before examining the final separation results, the change in the shape of the PV and HV histograms as the classification algorithm iterates are shown in Figure 27. This figure illustrates the histograms of the PV (column (c)), HV (column (b)) and their sum (column (a)) from the beginning to after the 3rd iteration (from the 1st to 4th row) and after convergence (the 5th row), respectively. From column (b) we can see that although the thresholds are selected optimally, the HV histogram before the 1st iteration is far from perfect. It spreads to the right and contains a lot of PV voxels. This is reduced through the iterations and the HV and PV clusters are well separated when convergence is reached.



(a) The PV+HV histogram (b) The HV histogram (c) The PV histogram

Figure 27. Histogram changes at different iterations

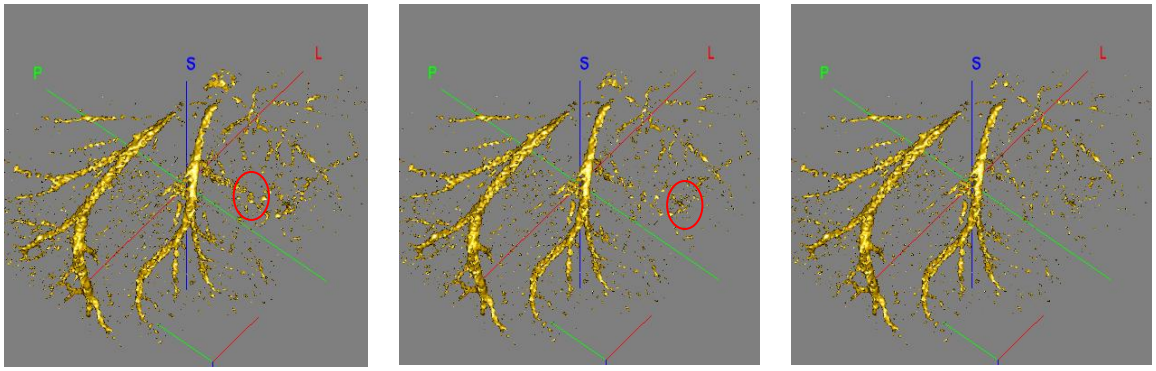


(a) The PV+HV histogram (b) The HV histogram (c) The PV histogram

Figure 27-- cont. Histogram changes at different iterations

Figure 28 shows the separated HV at different iterations in the same volume used to generate Figure 27. Results after the 1st iteration (Figure 28 (a)) contains a lot of PV

voxels, as shown by the red circle. This is consistent with the histogram shown in the 2nd row of Figure 27 (b), where the HV histogram tends to spread to its right side. The number of PV voxels is reduced in the HV result after the 3rd iteration (Figure 28 (b)), as its corresponding histogram contains less PV voxels (the 4th row of Figure 27 (b)). Figure 28 (c) shows the final HV after the iterative process converges, where most of the spurious PV voxels are removed.



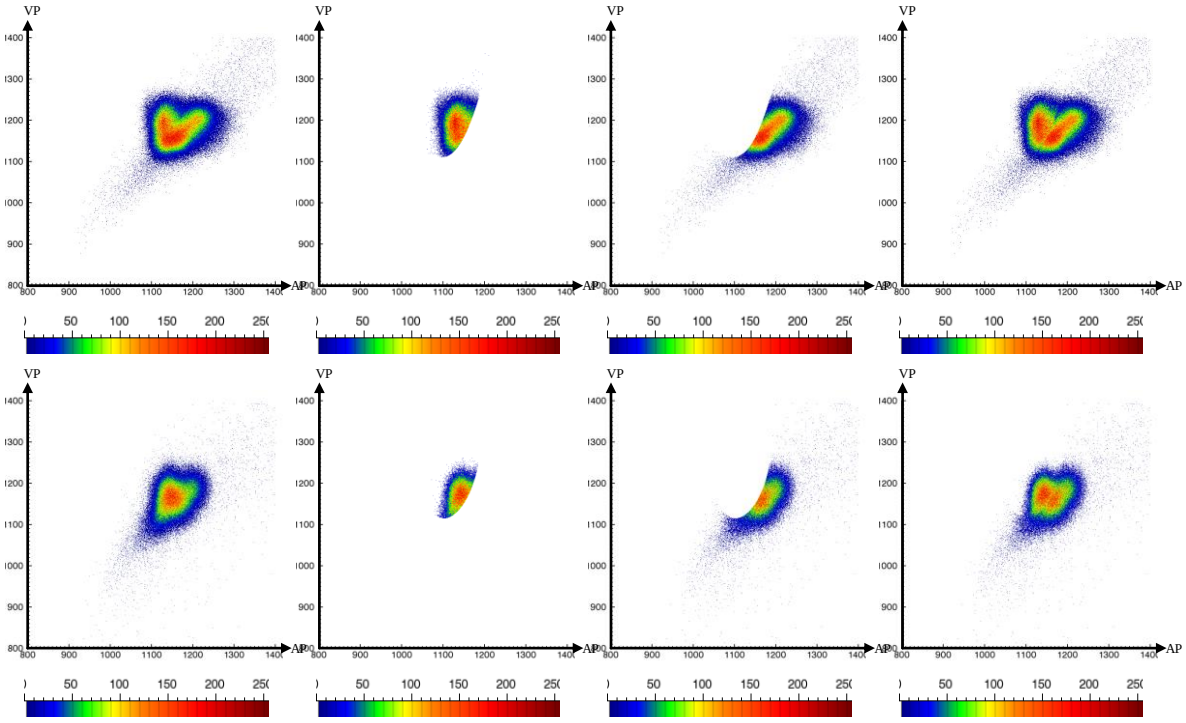
(a) The separated HV after the 1st iteration

(b) The separated HV after the 3rd iteration

(c) The separated HV after convergence

Figure 28 Separation results at different iterations

Figure 29 shows results obtained on two other data sets. The first column (Figure 29 (a)) shows the sum of the two initial PV and HV histograms. They overlap with each other, but one can visually draw a curve that could separate the two clusters. The second and third column (Figure 29 (b) and (c)) are the separated HV and PV histogram, respectively. The fourth column (Figure 29 (d)) shows the sum of the two separated histograms. This panel shows that the separation curve obtained with our method corresponds to the curve one could draw visually on the left panels to separate the two classes.



(a). The PV+HV histogram before separation

(b). The HV histogram after separation

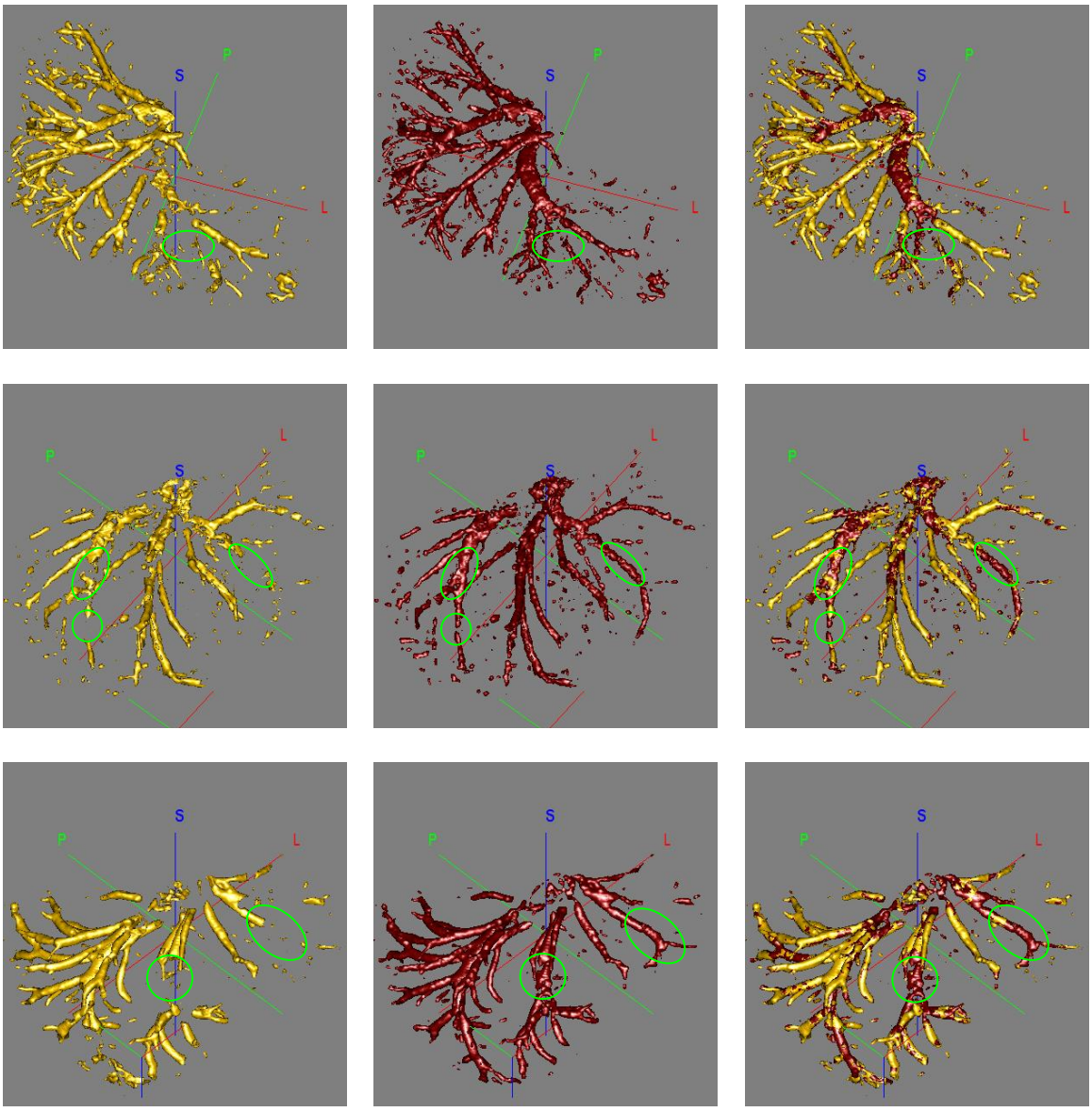
(c). The PV histogram after separation

(d). The PV+HV histogram after separation

Figure 29. Histogram analysis before and after separation

Figure 30 compares the HV separation results obtained by histogram analysis after Hessian enhancement, with those obtained with simple subtraction. The green circles points to regions of difference between the two results. The first column of Figure 30 shows results obtained with simple subtraction, which show a number of disconnected vessels. The second column shows separation results obtained with the iterative histogram classification, which fills in the disconnection and leads to more complete hepatic vessels. The third column shows both trees overlapped with each other, for better visual inspection and comparison. The red pixels in the third column represent those

pixels that are classified as HV using our improved approach but are missing if just a simple subtraction is used.



(a). Separated HV by subtraction

(b). Separated HV by histogram analysis

(c). Overlapping of (a) and (b)

Figure 30. Comparison of separated HV with Hessian enhancement by subtraction and histogram analysis

CHAPTER IV

HEPATIC VESSEL SEGMENTATION

IV.1 Introduction

In Chapter 3, we have discussed how to separate the portal vein and the hepatic vein in the venous phase image. This results in two images volumes (the separated PV and HV images), each of which contains only one vessel tree. In this chapter, we will discuss how the vessel trees can be segmented from these images.

Before segmentation, a median filter is applied to the hessian enhanced images. Median filters reduce speckle noise while preserving the edges in the original image. The Hessian filter we use to pre-process the images enhances tubular structures and weakens plate and blob like objects, but it cannot eliminate other artifacts, such as salt and pepper noise. Filtering the Hessian-filtered images reduces these artifacts and facilitates further processing.

The segmentation of the vessel tree in the separated images is a three-step iterative process. It involves a traditional region growing method, an adaptive directional region growing method, and a reconnection step. The traditional region growing procedure starts with an initial seed and a threshold. The algorithm is then applied to the entire image. The segmentation result is rough and higher order vessel branches may not be captured due to noise. But the main branches of the hepatic vasculature are well segmented and can be used as guidance for subsequent processing steps. The first of

these is skeletonization, which is used to extract the centerline of the detected vessels and to prepare for the next step: adaptive directional region growing.

In the proposed algorithm, a second region growing algorithm is used, which is confined to a local area, defined as a cylindrical region of interest (ROI) aligned with the detected vessel branches. The iterative procedure starts with initial seeds (which are the end points of the vessel skeleton detected in the previous step) and is limited to ROIs with predicted orientation. Segmentation is applied repeatedly to the local ROIs, whose information is stored in a queue, until no new voxel can be added to the segmentation results. The cylindrical ROI adapts its size, position and orientation to the predicted properties of the vessel branches to be segmented based on the segmentation result of its parent ROI. This idea is based on ideas introduced in [33] and [57]. But in those two papers the methods are applied to MRA head/neck and CT airway images. Here we have adapted the method to make it applicable to our CT hepatic images.

The third step is reconnection, which completes the vessel systems by re-attaching disconnected pieces that have not been included in the tree in the first two steps. It is a simpler version of the method used in [52]. Step 2 and step 3 are iteratively repeated until no more new vessel branch is found.

The following flow diagram (Figure 31) shows the overall scheme we use in our segmentation method.

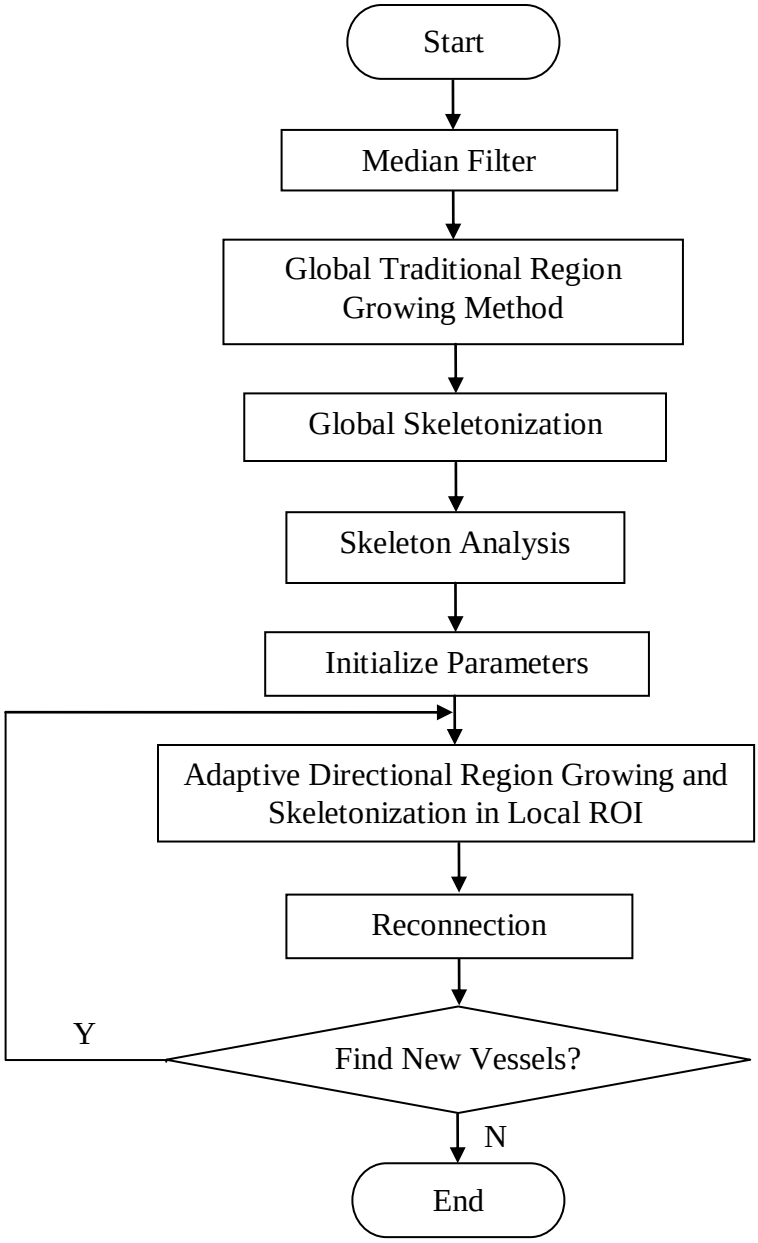


Figure 31. Flow chart of the proposed segmentation method

IV.2 Method

IV.2.1 Step 1: Traditional Region Growing Method

IV.2.1.1 Region Growing Method

The seed for this region growing method can be selected either manually or automatically, both of which approach are simple and easy to implement. Manual selection can be done by clicking on a point that belongs to the vessel using a mouse. Automatic selection can be implemented by thresholding the image with a much higher threshold than the one that is used to segment the vessels and then choose, as seed, the largest connected component in the binary image.

An example of traditional region growing result is shown below.

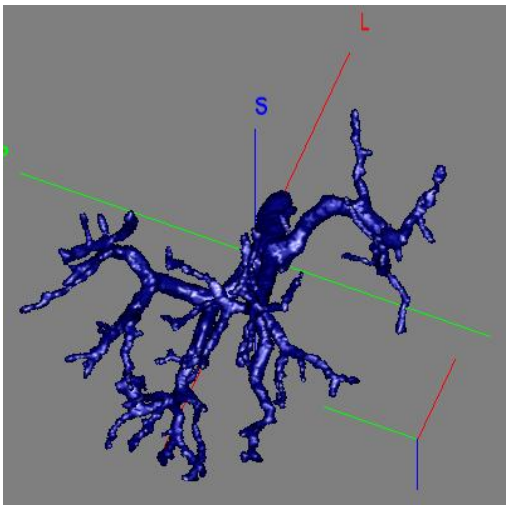


Figure 32. Segmentation result from traditional region growing method

IV.2.1.2 Skeletonization

As seen in Figure 32, the result obtained with a traditional region growing method is coarse. The main vessels are detected but a number of high order branches are missing and the surface of the vessels is bumpy. Such irregularities affect the results obtained with the skeletonization algorithm (cavities may lead to circles in the skeleton and convexities to spurious branches). This problem can be partly solved by applying morphological closing (a dilation followed by erosion with a certain structure element). The closing procedure before skeletonization fills small gaps and holes. Spurious branches caused by bumps on vessel surfaces can be removed by a pruning process after the skeletonization.

To perform skeletonization, which is the second step in our algorithm, we adopt the flux driven automatic centerline extraction algorithm proposed in [69] and [70]. In classical mechanics, the average outward flux of a vector field through the boundary of a region is negative if the region is shrinking under the Hamiltonian flow. In the limit, the average outward flux is close to zero if the region shrinks to a non-medial point, and results in large negative value if the region shrinks to a medial point. Based on this knowledge, a homotopology preserving thinning process is developed. This method removes simple points in descending order of the average outward flux of the gradient vector field (distance map) without altering the object's topology. When the average outward flux of a point is below a threshold, that point is removed only if it is not an end point. In this way, points on the border of the object are removed first, and points inside the object can only be removed when they become border points. The object is thinned

gradually, and finally the centerline is obtained. Pruning is performed to eliminate suspicious branches.

The average outward flux of the gradient vector field of the Euclidean distance function measures the likelihood of a voxel to belong to the centerline of an object. The more negative the value is, the more likely it is a skeletal point; the closer to zero the value is, the less probable it is that the point is on the centerline. Numerically, the average outward flux at voxel x is expressed as follows:

$$AOF(x) = \frac{1}{n}\sum_{i-1}^{n}\left\langle \hat{N}_i, \nabla D(x_i) \right\rangle \tag{13}$$

where x_i is an n-neighbor (n= 8 in 2D and n=26 in 3D) of x, $\hat{N}_i$ is the outward normal of the unit circle in 2D and sphere in 3D centered at x, and $\nabla D(x_i)$ is the gradient vector field of the distance transform of the object.

This thinning process has two parameters: the average outward flux threshold below which the end points are being kept in the skeleton instead of being removed, and the length threshold for pruning spurious branches. In our implementation, the flux threshold has been selected as a value such that 70% of the average outward flux is less than it. A branch is regarded as a spurious branch if the skeleton length is less than 3 pixels. This method works properly on our data set as shown in the figure below.

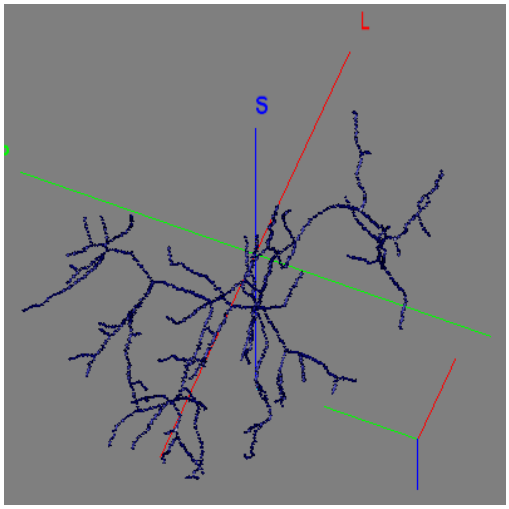


Figure 33. Skeleton of the traditional region growing result

IV.2.1.3 Skeleton Analysis and ROI Queue Establishment

The resulting skeleton is a 26-connected 3D curve in a cubic lattice which is no more than one voxel thick except possibly at branch points. Each point on this curve can be labeled by examining its 26-neighborhood and three types of points can be identified (Figure 34): (1) end points have only one neighbor, (2) curve points have exactly two neighbors and (3) branch points have at least 3 neighbors.

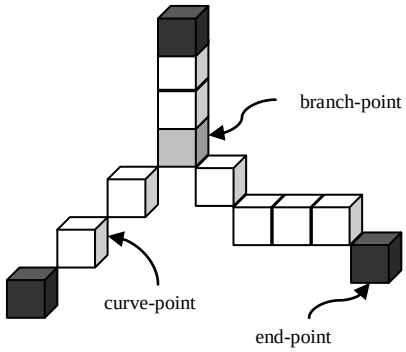
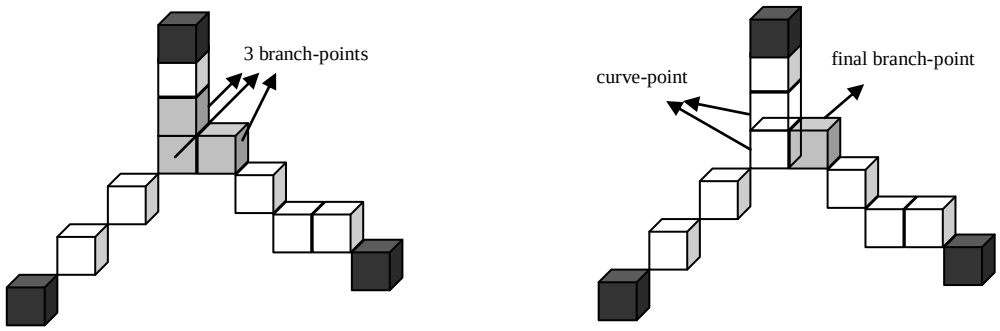


Figure 34. Three types of points in a skeleton

Branch point identification from a thinned and complicated tree is not a trivial problem because various situations can be encountered and a single criterion cannot handle all the cases. One problem is that more than one branch points may form a junction, as shown in the Figure 35. In this case, we just choose one of the branch points randomly and label it as a branch point. We then find all the other branch points which are 26-connected with that final branch point (that is, within a distance of $\sqrt{3}$) and label them as curve points.



(a). Multiple voxels in skeleton for one branch point (b). Branch point identification

Figure 35. One case of branch points in skeleton

IV.2.2 Step 2: Adaptive Directional Region Growing Algorithm

The adaptive directional region growing algorithm is applied at the end of each branch that has already been segmented. This is an iterative process and further growing along a given vessel branch depends on the previous vessel segment. We establish a queue to store the necessary information for growing and to simplify the iterative process. This ROI queue contains seed points and the growing directions, as well as the criteria for stopping the growing process. The end points of the skeleton tree are used as seeds for

adaptive directional region growing and as the starting position for the ROIs. The end points together with the corresponding branch points determine the orientation of the ROI inside which local segmentation is applied. At each iteration, a set of parameters are retrieved from the top of the queue, a new ROI is built based on the set of parameters, segmentation is performed within the ROI, and a new set of parameters is pushed into the bottom of the queue to be used to segment the child of the current vessel segment, if there is any.

This step is the core of the whole process and it is shown in details in the following flow chart:

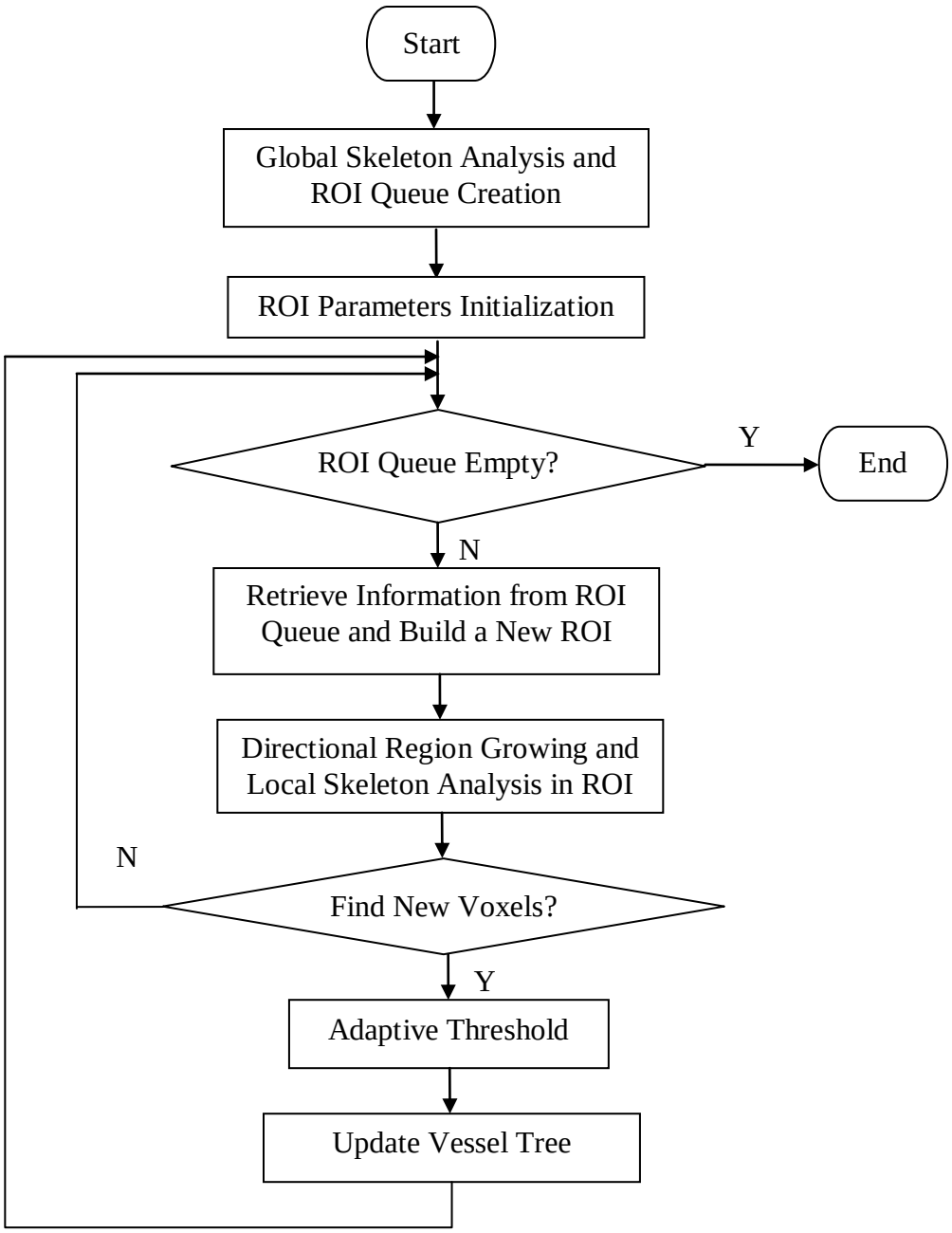


Figure 36. Flow chart of the local region growing segmentation

IV.2.2.1 Positioning New ROI

The proposed method uses cylindrically shaped ROIs [33], which are different from the more commonly used cube ROI [57]. Cylinders have been chosen because they

are a better fit for vessel branches than cubes. As a result, the search region is usually smaller and contains less noise, which improves the results. The inputs to the function used to place a new ROI include four parameters: the starting position, the orientation, the length, and width of the ROI. The new ROI starts from the end point of the skeleton of the parent branch. The orientation of the new ROI is determined by the direction of its parent skeleton, which is defined either by one end point and its nearest branch point or by two end points if the parent skeleton has no bifurcations. The ROI length and width are currently constant.

Before placing the new ROI along the predicted direction of the vessel segment, a reference cylindrical ROI is built. This ROI has the same starting point, same length and width as the predicted one, but the orientation is fixed: it is along the y axis (the positive direction). The rotation axis and rotation angle between the reference and final cylindrical ROI are calculated and thus the rotation matrix is computed and applied to the reference ROI. Thus the final ROI is aligned in the right direction. A local region growing procedure is performed inside this ROI. The figure below shows a sketch that describes this procedure.

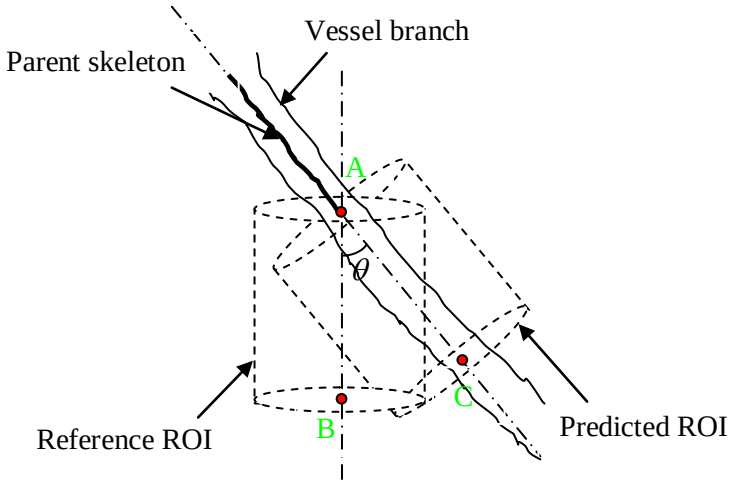


Figure 37. Region of interest (ROI) establishment

In this figure, A is the end point of the parent skeleton, vector AB is the cylindrical axis of the reference ROI, and vector AC is the axis of the predicted ROI, which is on the extended line of the parent skeleton. The rotation axis $\vec{\varpi}$ and rotation angle θ about that axis are computed as follows:

$$\theta = \cos^{-1}\left(\frac{AB \bullet AC}{|AB| \cdot |AC|}\right) \tag{14}$$

$$\vec{\varpi} = \frac{AB \times AC}{|AB| \cdot |AC| \cdot \sin\theta} \tag{15}$$

Once $\vec{\varpi}$ and θ have been determined, the rotation matrix R is calculated as a function of θ and $\vec{\varpi}$:

$$R = \begin{bmatrix} \varpi_x^2 V + C & \varpi_x \varpi_y V - \varpi_z S & \varpi_x \varpi_z V + \varpi_y S \\ \varpi_x \varpi_y V + \varpi_z S & \varpi_y^2 V + C & \varpi_y \varpi_z V - \varpi_x S \\ \varpi_x \varpi_z V - \varpi_y S & \varpi_y \varpi_z V + \varpi_x S & \varpi_z^2 V + C \end{bmatrix} \tag{16}$$

where the ϖ_x, ϖ_y, ϖ_z are the three components of $\vec{\varpi}$, $V = 1 - \cos\theta$, $C = \cos\theta$, and $S = \sin\theta$ [71].

Two particular angles should be treated with caution, 0° and 180°, which means no rotation at all and a rotation of 180°, respectively. When the angle is very close to 0° or 180°, the rotation matrix is set to the identity matrix $R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ or $R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$, respectively, instead of computing the rotation axis to avoid "NaN" (Not a number) caused by numerical errors.

IV.2.2.2 Local Region Growing

After median filtering, some of the vessel branches appear as several disconnected pieces. Conventional region growing algorithms stop at the disconnected place. If we could find those disconnected parts that belong to the same vessel branch (the disconnected parts lying on the extended line of the vessel branch) and link them together, we then could find more branches than the conventional region growing. To achieve this, we detect all regions in the cylindrical regions of interest made of pixels that are within an intensity range. We keep the region that is 26-connected with the seed point as well as the first and second largest connected components within the ROI. These regions are then considered as candidate branch segments and their skeleton is extracted. Next, the angle these segments make with the direction of the parent vessel branch is computed and if this angle is below 30°, the segment is connected to the parent branch.

In our method, an adaptive threshold instead of a fixed one is used to do the segmentation. The segmentation result in the current ROI is used to estimate the threshold for the segmentation in its child ROI. After local region growing in the current ROI, if new vessel voxels have been found, the intensity histogram of the vessel segment is computed, then a Gaussian curve is used to fit the histogram and its mean and the standard deviation are estimated. The intensity threshold is set as the mean minus one standard deviation. For local region growing in ROIs occurring right after the traditional region growing (the first iteration), an initial threshold is provided.

IV.2.3 Step 3: Reconnection

At branch junctions, the response of the Hessian filter is relatively low compared to its response at other locations. This is so because the assumption of tubular structure is violated. As a result, some branches are isolated from the main trunk. In these cases, the disconnected piece cannot be found and linked to the main vessel tree by the adaptive directional region growing approach introduced in the previous section; this is so because these disconnected segments do not fall in ROIs extending the current vessel tree. A reconnection algorithm described below is used to recover some of these segments.

Our assumption for reconnection is that the break point at which the branch is disconnected from the main trunk is very close to the actual branch junction. So a reconnection is carried out only if the component has a distance less than a threshold T_dist from the main vessel tree. This is implemented as follows:

1. Everywhere in the image, select those voxels that have a distance smaller than T_dist but greater than 0 from the main trunk and have intensities greater than a threshold, T_intensity, as seeds for region growing.
2. After region growing, connected components with a voxel number greater than a count threshold T_num and not contained in vessels that are already segmented are regarded as candidate branches.
3. Local skeletonization is applied to the candidate branches.
4. A cylindrical searching area with length L_1 along the direction of the candidate branch, started from the reconnection point (the end point of the local skeleton that is the closest to the main vessel skeleton) is defined. Inside the searching area, we find a reconnection path which is the shortest between the reconnection point and the main vessel skeleton tree (Figure 38).

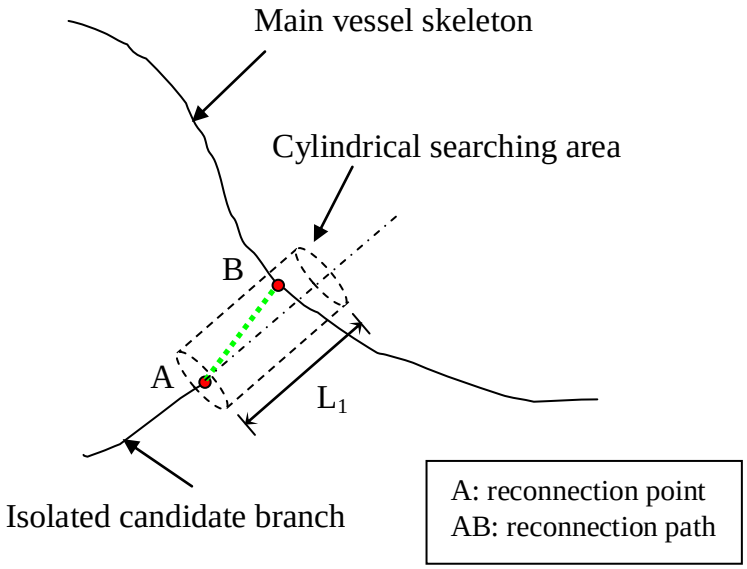


Figure 38. Vessel reconnection in cylindrical searching area

The two steps, adaptive directional region growing and reconnection are alternated and repeated to complete the vessel tree by adding possible candidate branches to the main trunk. This iterative process can be stopped using various criteria. For example, the process could stop when no new vessel segment is detected.

IV.3 Results and Discussion

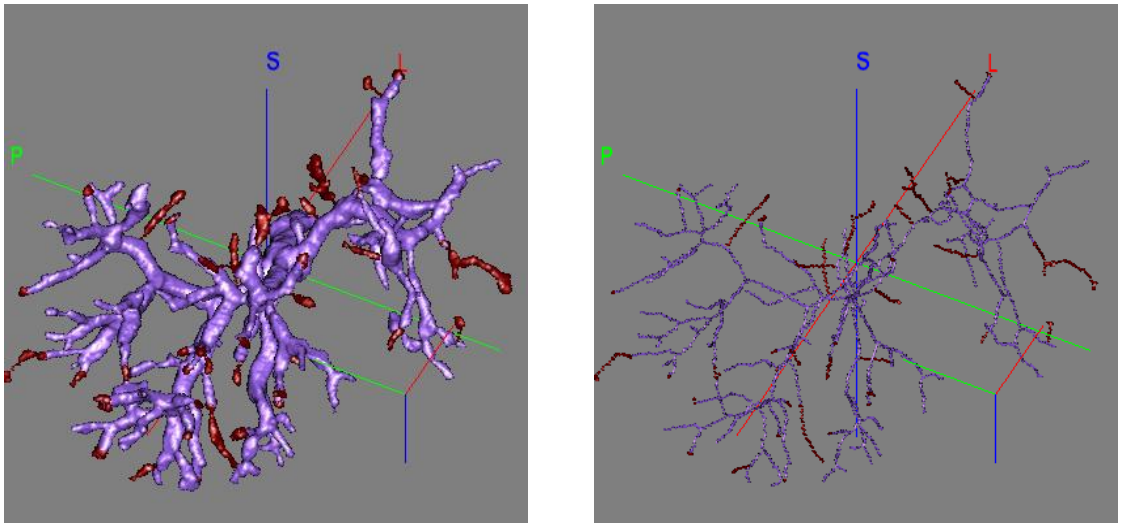
IV.3.1 Different Schemes for Portal Vein Segmentation

Since both the arterial and the venous phase images contain the portal vein, the portal vein could be segmented using either of these images (after separation in the venous phase image) or using both. In the following sections we will present results obtained with various approaches. In this discussion, we will assume that the arterial phase image has been registered non-rigidly to the venous phase image, as discussed earlier.

IV.3.1.1 Portal Vein Segmentation in the Arterial Phase Image

Because the arterial phase image includes the portal vein only, the most straightforward segmentation scheme is to extract the portal vein from this image volume. As described in the previous sections, traditional region growing is first applied, followed by directional region growing, and reconnection. Results obtained with this approach are shown in Figure 39. To permit comparison, the results obtained with our method are shown next to the results obtained with a traditional region growing technique. Vessels/skeletons shown in purple are obtained with a traditional region growing

algorithm, while those shown in red are the additional vessels/skeletons we have detected with our method.



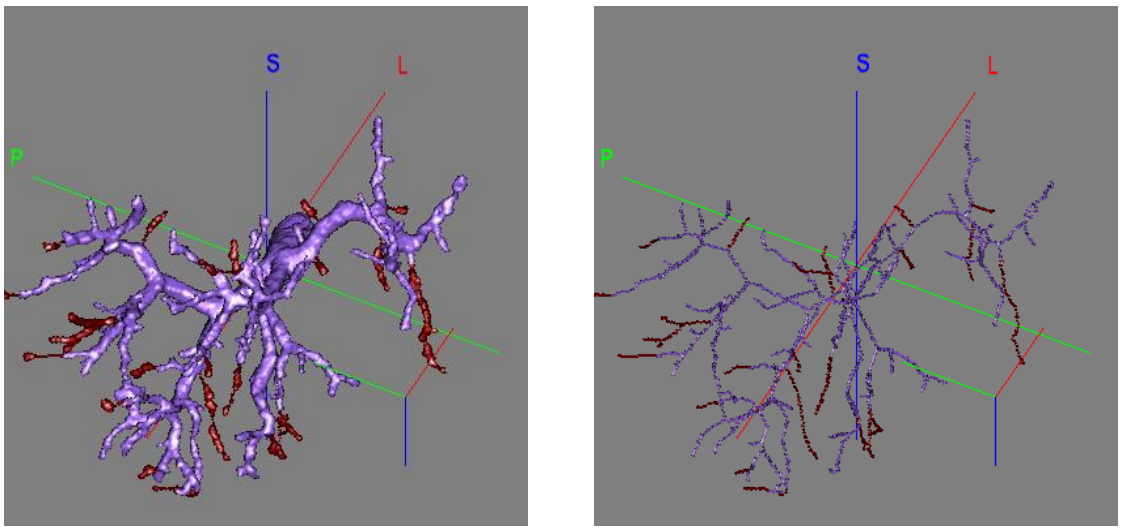
(a). The segmented portal vein (b). Skeleton of the portal vein

Figure 39. Portal vein segmentation results from the AP image

The segmentation and skeletonization results for the portal vein shown above, illustrate that our method can extract and connect more higher order vessel than a standard region growing technique.

IV.3.1.2 Portal Vein Segmentation in the Separated Venous Phase Image

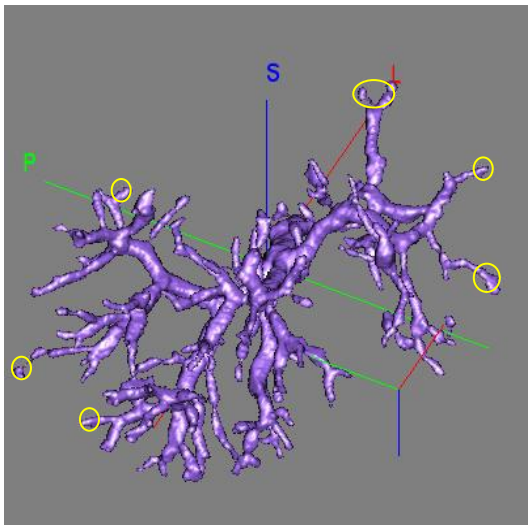
As is the case for the arterial phase image, the separated PV image only contains the portal vein. The final portal vein segmentation and skeletonization results obtained with our method on the separated venous phase image corresponding to the arterial phase image used above are shown in Figure 40. Comparing these with the results obtained with the traditional region growing method, we can see that the proposed approach outperforms its counterpart by finding more small and/or isolated vessel branches.



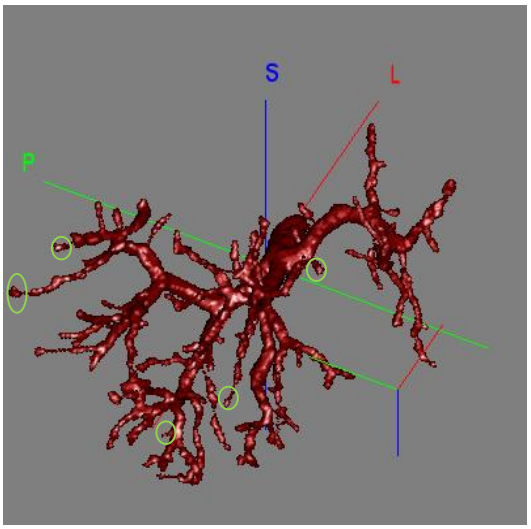
(a). The segmented portal vein (b). Skeleton of the portal vein

Figure 40. Portal vein segmentation results from the VP image

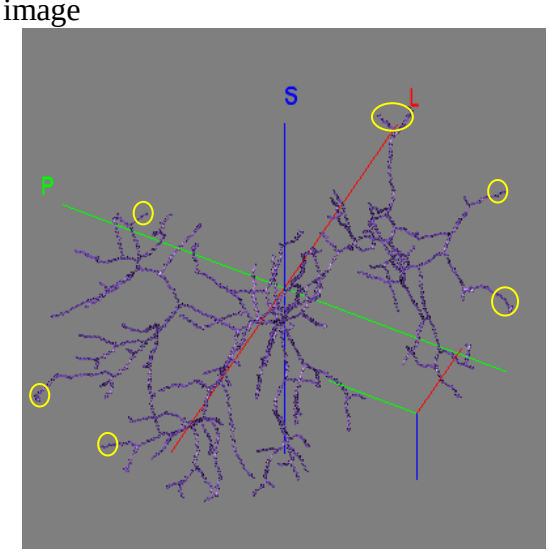
Figure 41 shows the segmented portal vein and its skeleton obtained from the AP image (shown in purple) and the VP image (shown in red) side by side. The yellow circles and green circles label the additional vessels/skeletons found in one phase but missed in the other. This figure shows that each phase contains some exclusive PV information, so the PV segmentation achieved using information from both phases could be better than that using just one phase.



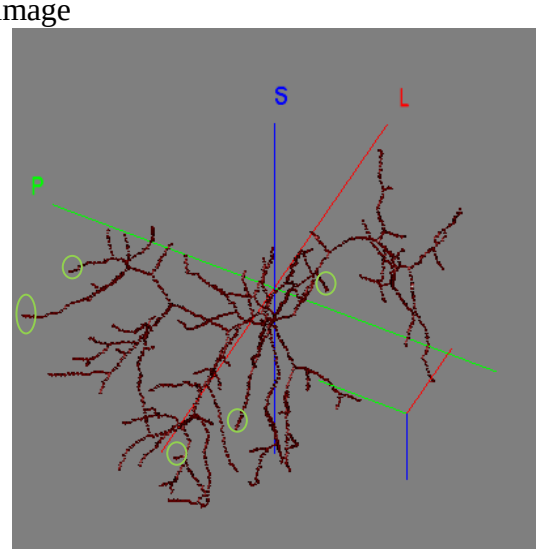
(a). Vessel segmentation from the AP image



(b). Vessel segmentation from the VP image



(c). Vessel skeleton of (a)



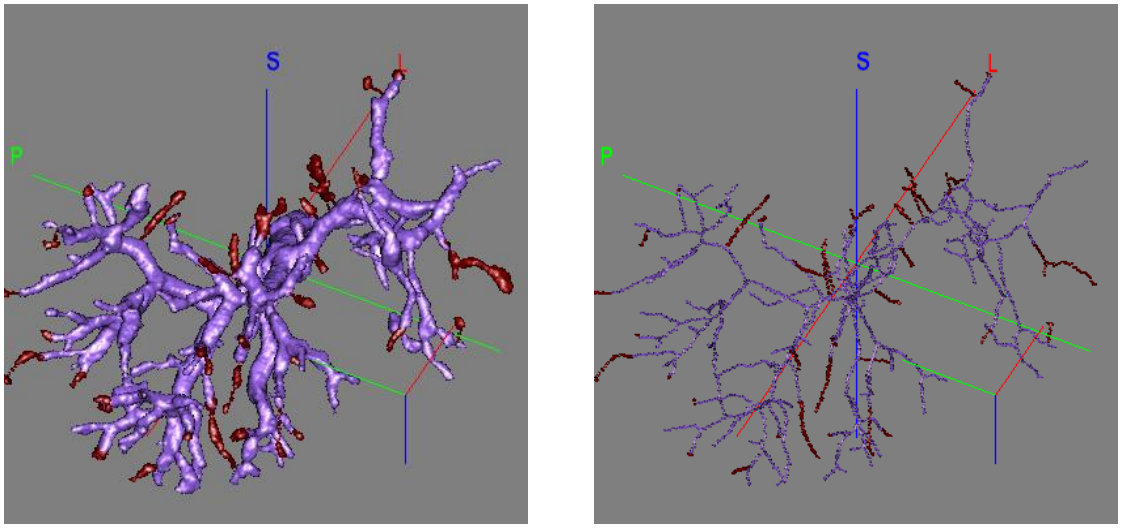
(d). Vessel skeleton of (b)

Figure 41. Portal vein segmentation results comparison: segmentation in the AP image vs. segmentation in VP images

IV.3.1.3 Portal Vein Segmentation using both the Arterial and the Venous Phase Images

In this scheme, we first apply the regular region growing algorithm on both the arterial and venous phase images separately, which produces two initial trees. These two trees are then combined and the skeleton of the combined tree is extracted. This skeleton

is used to initialize the subsequent segmentation steps in both the arterial and venous phase images, again separately. Next, the final trees obtained in each phase are combined to produce the final results. Figure 42 shows the segmented portal vein and its skeleton obtained using information from both phase images. Again, our results are better than those obtained with a standard region growing method.

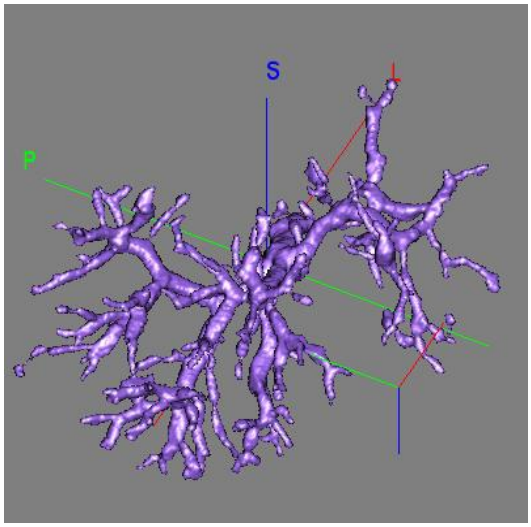


(a). The segmented portal vein (b). Skeleton of the portal vein

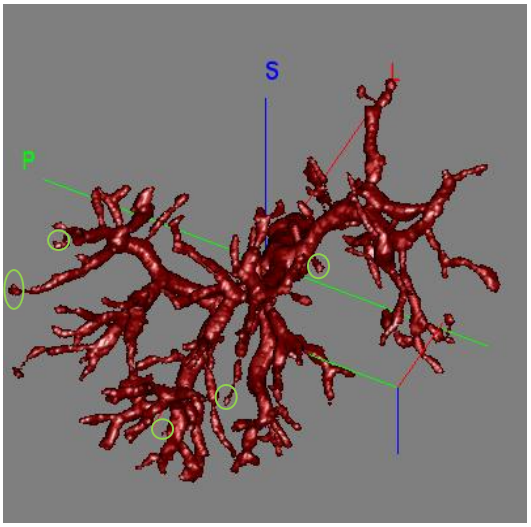
Figure 42. Portal vein segmentation results obtained with both the AP and VP images

Figure 43 compares the portal vein and its skeleton obtained from the scheme described above (shown in red) with those obtained from the AP image alone (shown in purple). Figure 44 shows the differences between the results of this scheme (shown in red) and those of the segmentation in the separated PV image of the venous phase alone (shown in purple). The results obtained with one phase alone and with both phases are shown side by side, and the differences are labeled with green circles. These results show that the sum of the segmentation results from both image phases are better than those

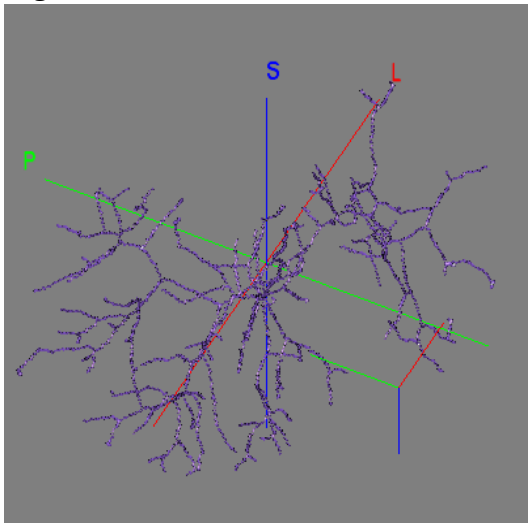
from either phase alone. This is so because the PV information in one phase can complement the information provided by the other.



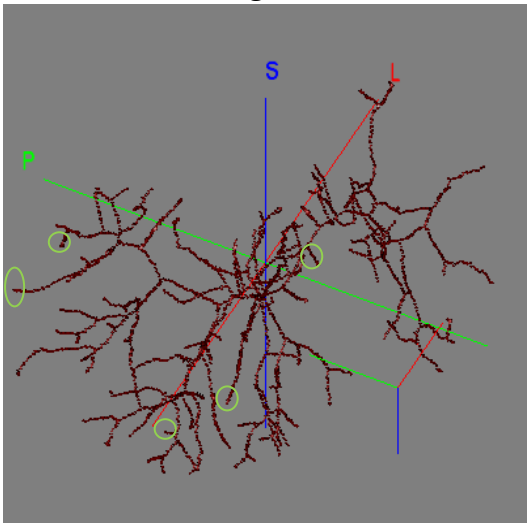
(a). Vessel segmentation from the AP image



(b). Sum of the vessel segmentation from both AP and VP images

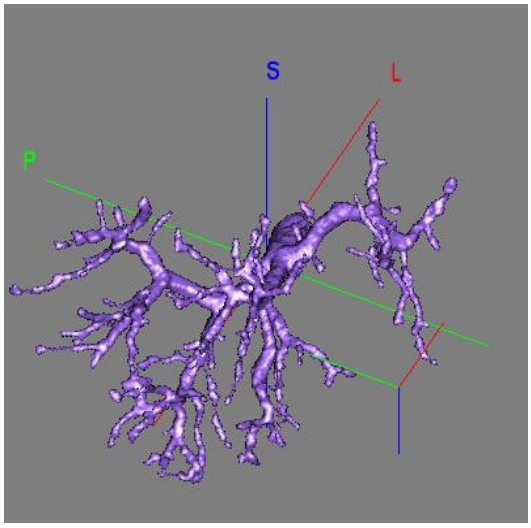


(c). Vessel skeleton of (a)

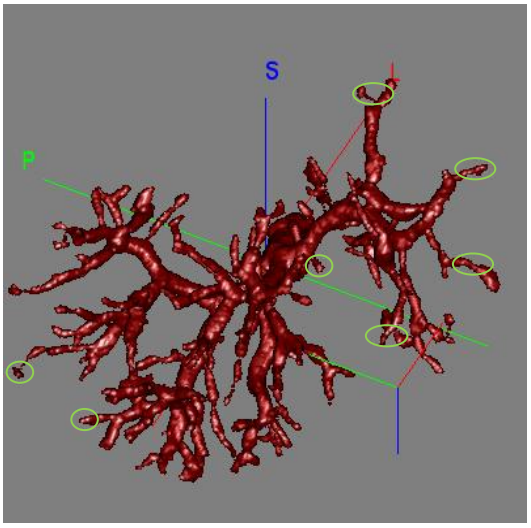


(d). Vessel skeleton of (b)

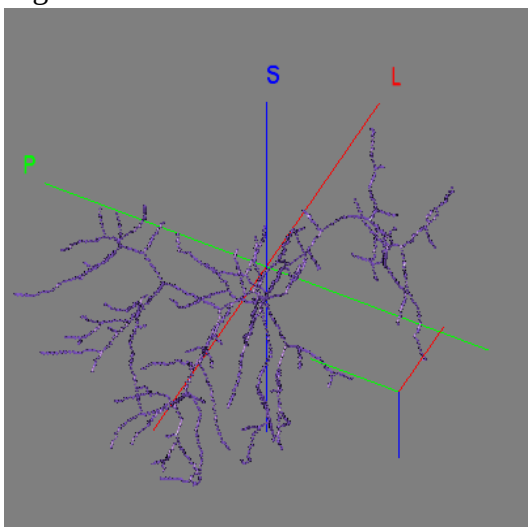
Figure 43. Portal vein segmentation results comparison: segmentation in the AP image alone vs. sum of the segmentation in the AP and VP images



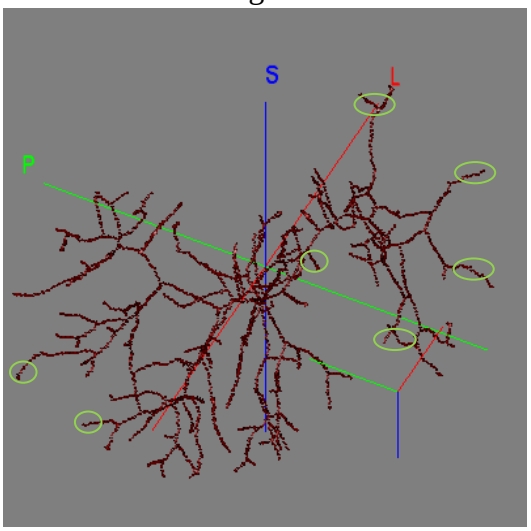
(a). Vessel segmentation from the VP image



(b). Sum of the vessel segmentation from both AP and VP images



(c). Vessel skeleton of (a)

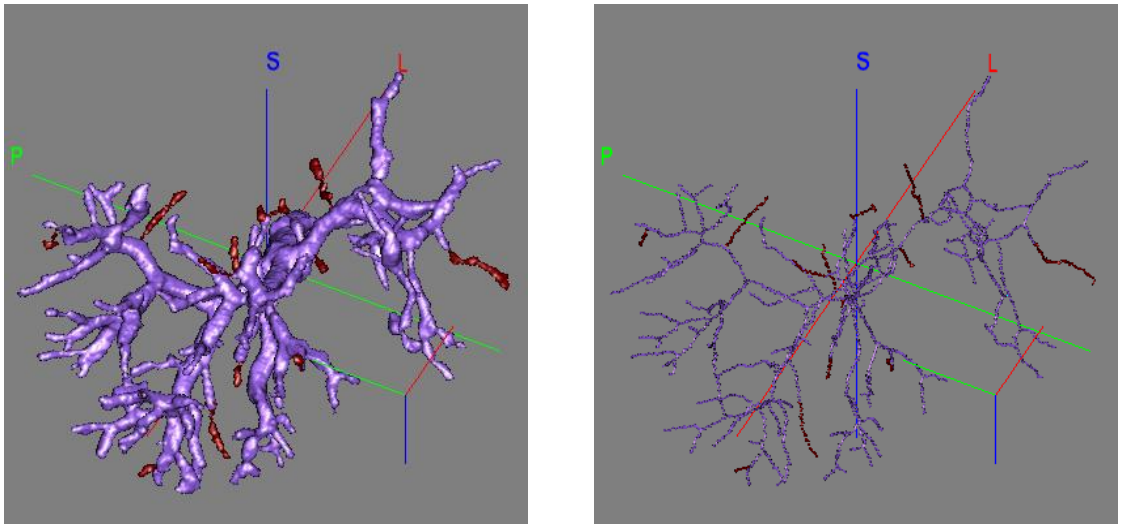


(d). Vessel skeleton of (b)

Figure 44. Portal vein segmentation results comparison: segmentation in the VP image alone vs. sum of the segmentation in the AP and VP images

IV.3.1.4 Portal Vein Segmentation with an Averaged Arterial and Venous Phase Image

This scheme is similar as the previous one in IV.3.1.2 for the initial vessel segmentation and skeleton extraction, except that, instead of performing the segmentation and reconnection approach in both phases and then sum the results, it is applied to the average of the two phase images. Figure 45 shows the results of this scheme compared with those from a regular region growing technique.

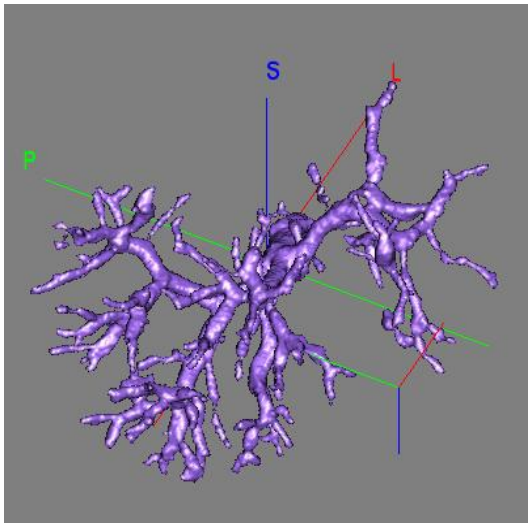


(a). The segmented portal vein (b). Skeleton of the portal vein

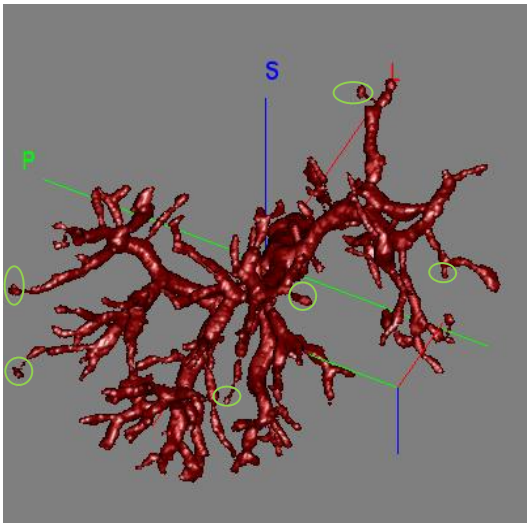
Figure 45. Portal vein segmentation in the averaged AP and VP images

In Figure 46, the portal vein and its skeleton obtained with this scheme (shown in purple) are compared with the results obtained with the scheme described in IV.3.1.2 (shown in red). The latter outperforms the former by finding more detailed vessel branches, as shown with the green circles. This can be explained as follows: each image phase may contain some unique PV information. After averaging, such information could be attenuated, or even be lost, so the resulting vessel is less complete than the one

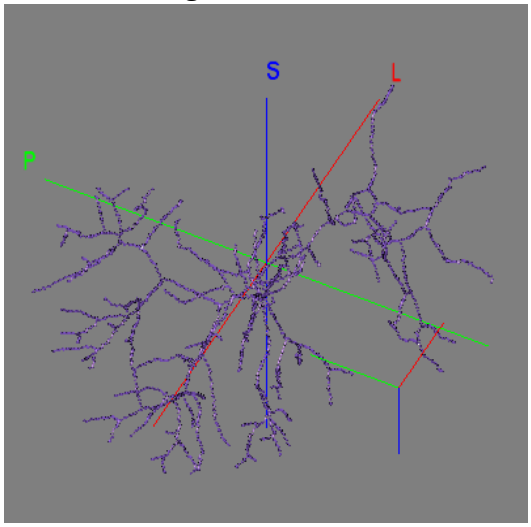
obtained by summing the segmentation results obtained with the AP and VP images, which preserves the unique PV information in each phase.



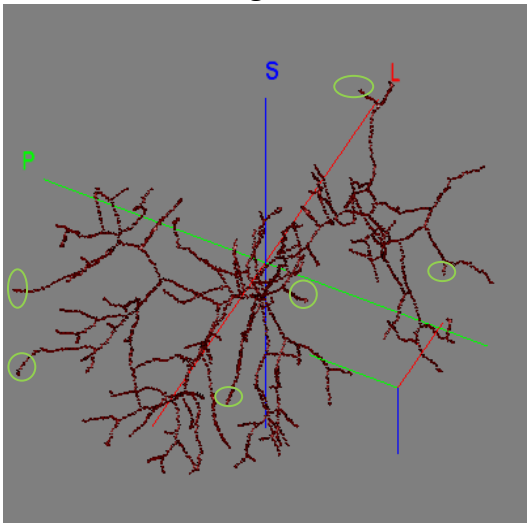
(a). Vessel segmentation from the averaged AP and VP images



(b). Sum of the vessel segmentation from both AP and VP images



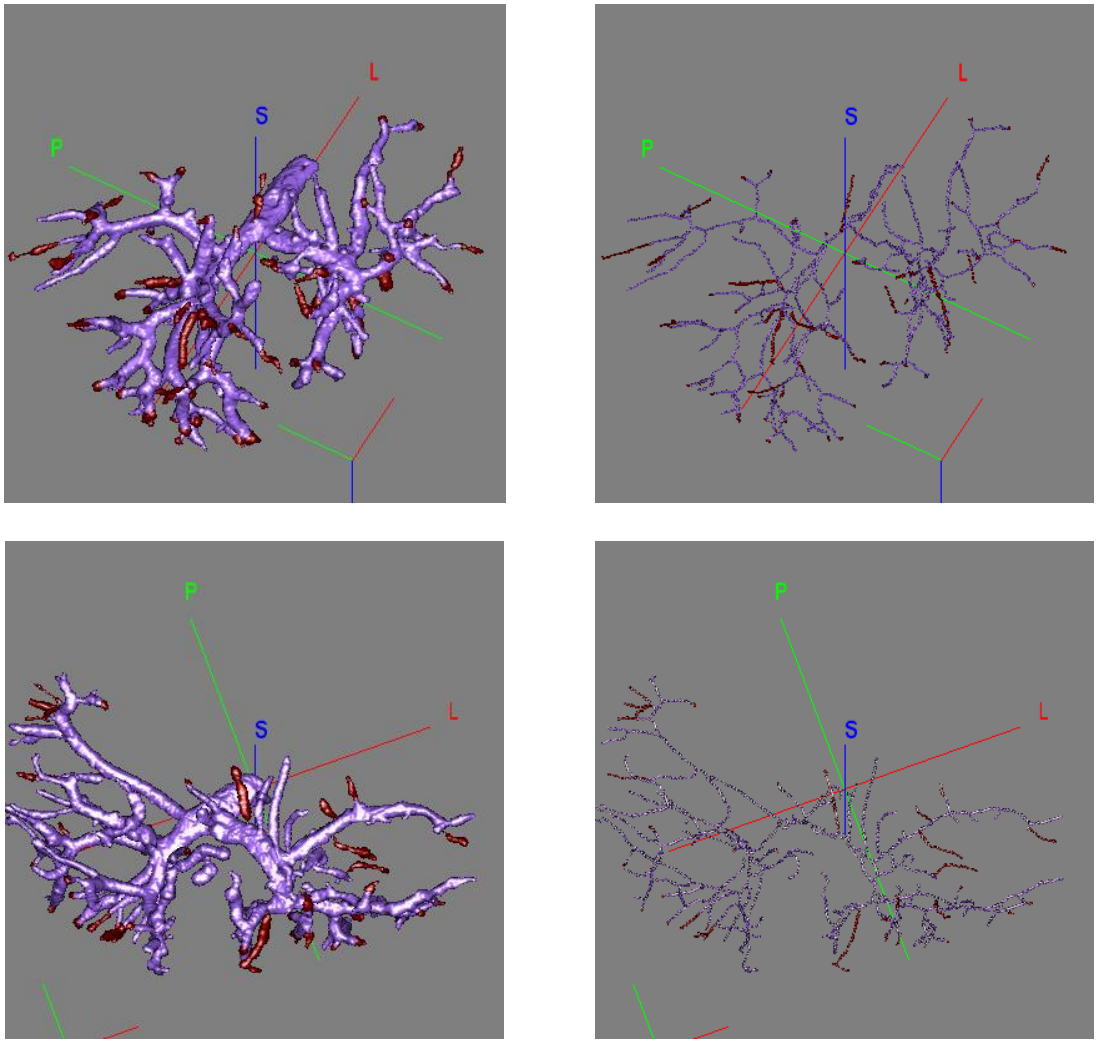
(c). Vessel skeleton of (a)



(d). Vessel skeleton of (b)

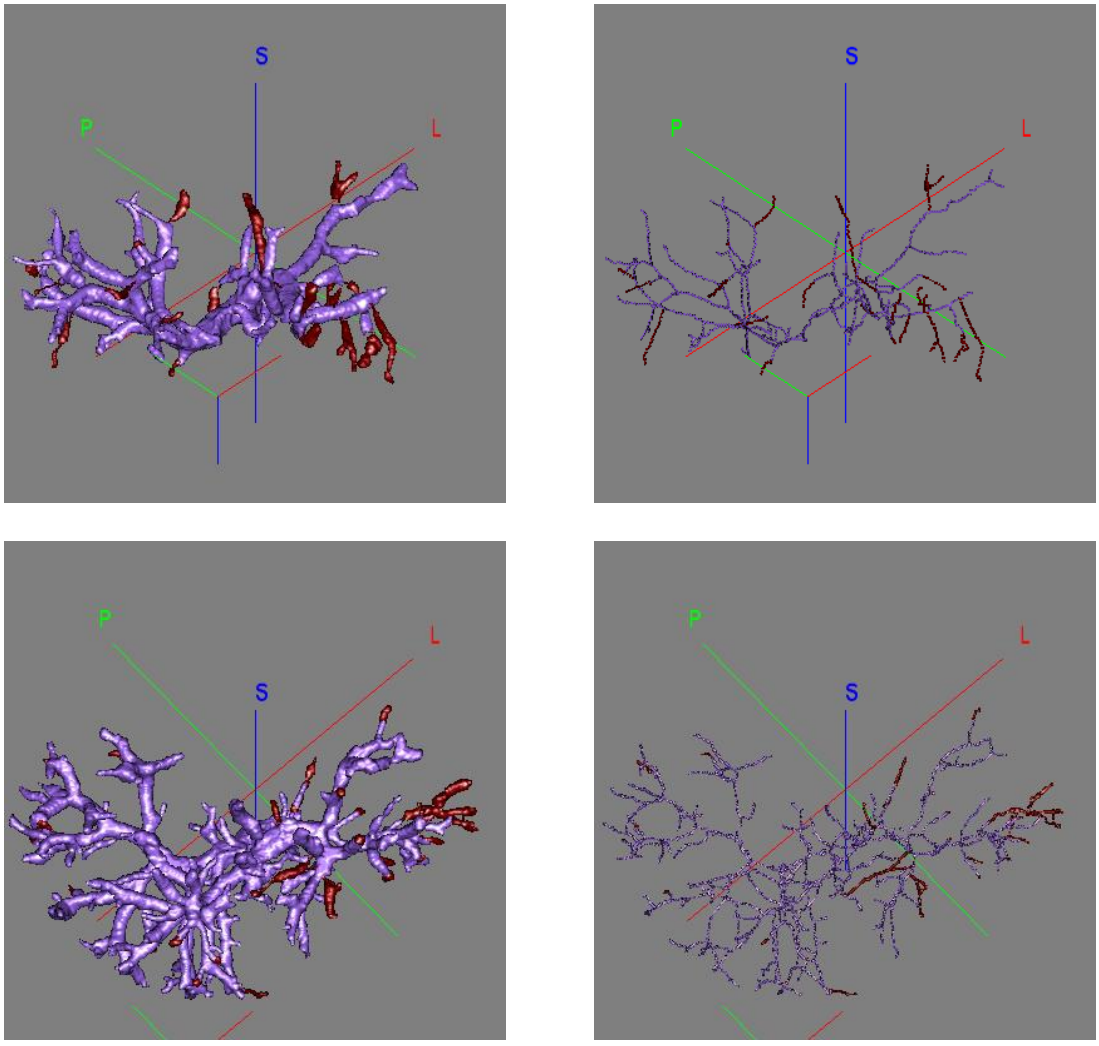
Figure 46. Portal vein segmentation results comparison: segmentation in the averaged AP and VP images vs. sum of the segmentation in the AP and VP images

By comparing the different portal vein segmentation schemes described above, we conclude that the one that combines the segmentation results from both the AP and PV images performs the best. Figure 47 presents the results obtained with this segmentation scheme (shown in red) obtained on the other data sets used in our study. It also shows the results obtained with a standard region growing method (shown in purple) for comparison.



(a). Segmented portal vein (b). Skeleton of the portal vein

Figure 47. Portal vein segmentation and skeletonization results



(a). Segmented portal vein (b). Skeleton of the portal vein

Figure 47 -- cont. Portal vein segmentation and skeletonization results

There are several parameters used in the iterative segmentation approach, some of which need to be adjusted because of large differences between data sets. Here is the list of the parameters used and how we choose them for our current study:

Table 4. Parameters used in the segmentation method

Parameters	Explanations	Selecting Rules
T_rg	The intensity threshold for the standard region growing method.	Need adjustment. Set as the lowest intensity that does not produce incorrect result based on visual inspection. Select a value, run the standard region growing algorithm, and check if the resulting vessel branches form circles, are connected to spurious points, or leak into liver parenchyma. If so, increase the value by 5; if not, lower the value by 5; repeat the region growing, until a vessel tree without holes or spurious branches is found.
T_rg_initial	The initial threshold for the adaptive directional region growing.	Need adjustment. Usually a little bit lower than T_rg, since the intensities at higher order vessel branches are lower than those at the main vessel. Set initially as 2/3T_rg, then round the value to the nearest number which is a multiple of 5. Run the adaptive directional region growing segmentation, and check the vessel segments that are newly grown. If they contain noise or leak into liver parenchyma (usually with irregular shape), increase the value by 5 and repeat the segmentation, until clear and elongated vessel segments are found, if there are any.
ROI_width	The width of the region of interest for the adaptive directional region growing.	Fixed. ROI_width=10
ROI_length	The length of the region of interest for the adaptive directional region growing.	Fixed. ROI_length=20
T_dist	Used in the reconnection, assuming the detached pieces have a distance less than T_dist from the main vessel.	Need adjustment. Default value is 10. After the reconnection step, check the vessel segments that are newly attached. If they contain noise or leak into liver parenchyma (usually with irregular shape), decrease the value by 2, and repeat the segmentation, until clear and elongated vessel segments are found, if there are any.

Table 4 -- cont. Parameters used in the segmentation method

T_intensity	Used in the reconnection step, assuming the detached pieces have an intensity greater than T_intensity.	T_intensity=T_rg.
T_num	Used in the reconnection step, assuming the detached pieces have a voxel number larger than T_num.	Need adjustment. Default value is 50. After the reconnection step, compare the result with the Hessian enhanced image, if there are lots of small disconnected pieces that have not been attached to the main vessel tree, lower the value by 10, and repeat the reconnection step, until most of the pieces are found.
L1	The length of the cylindrical searching area in reconnection.	Fixed. L1=30

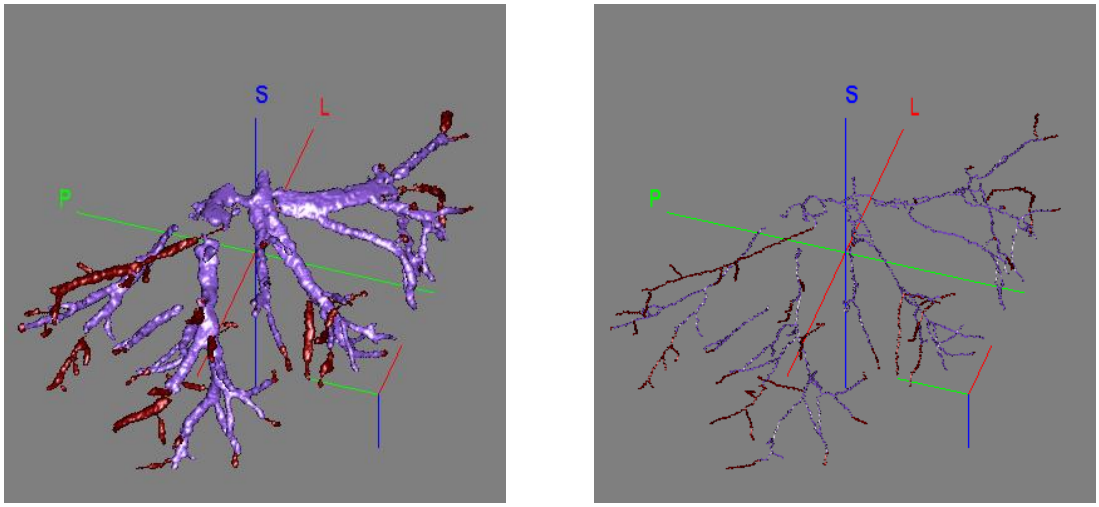
Table 5 shows the values of the parameters that need to be adjusted, that were used to obtain the results shown in Figure 42 and 47.

Table 5. Values of the parameters that need to be adjusted

Parameters	Data set 1	Data set 2	Data set 3	Data set 4	Data set 5
T_rg	20	60	40	30	50
T_rg_initial	15	40	30	25	40
T_dist	10	6	6	6	10
T_intensity	20	60	40	30	50
T_num	20	50	50	50	50

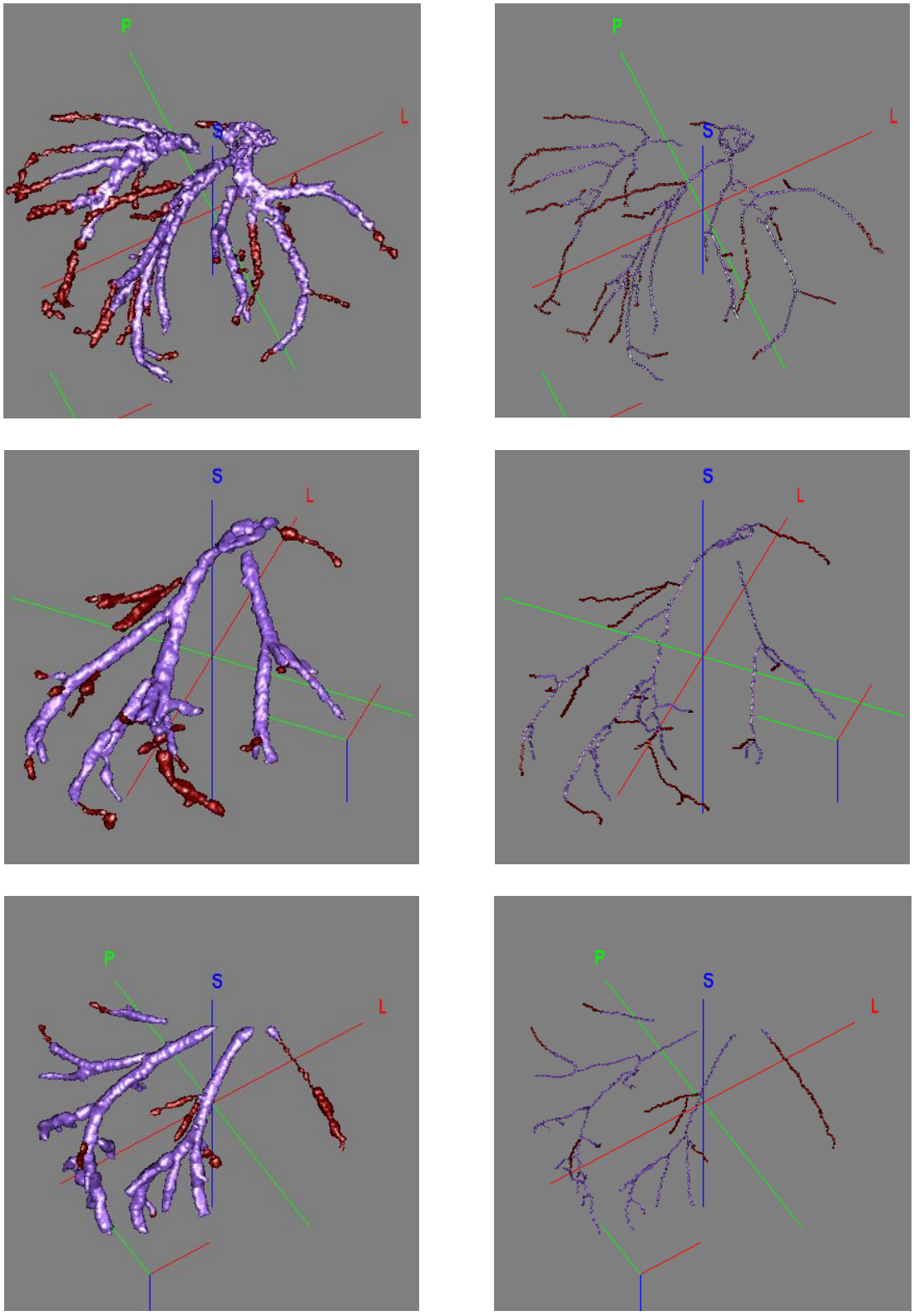
IV.3.2 Vessel Segmentation of the Hepatic Vein

Since the hepatic vein appears only in the venous phase image, segmentation of the hepatic vein is more straightforward. The initial vessel and skeleton are extracted from the separated HV image of the venous phase, and the following segmentation steps are performed in the same image. Figure 48 shows the segmented hepatic veins and their skeletons obtained with our approach (shown in red). It also compares them with the traditional region growing results (shown in purple). Again, our approach not only can find smaller vessel branches, but it also can recover branches detached from the main vessel tree.



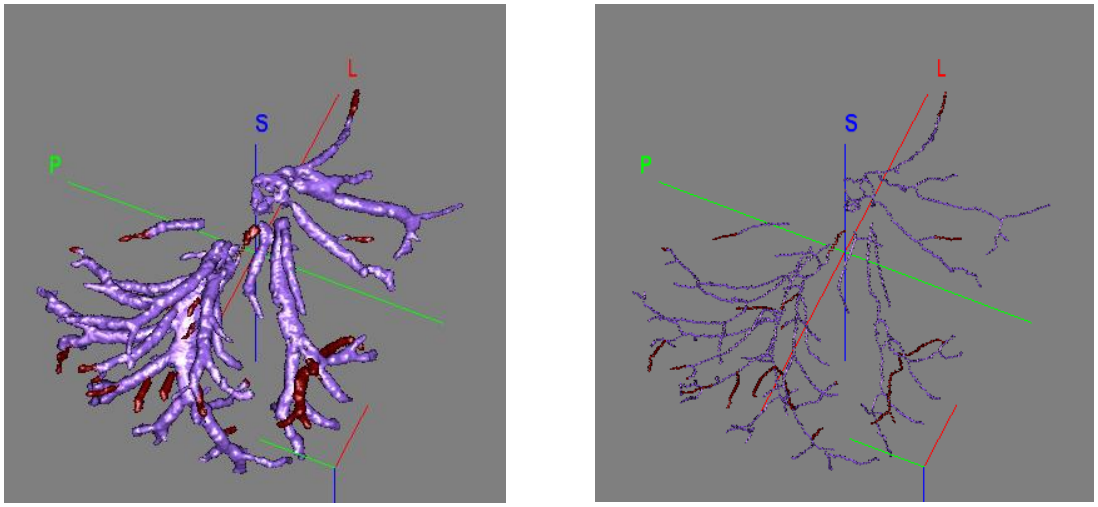
(a). Segmented hepatic vein (b). Skelton of the hepatic vein

Figure 48. Hepatic vein segmentation and skeletonization results



(a). Segmented hepatic vein (b). Skelton of the hepatic vein

Figure 48 -- cont. Hepatic vein segmentation and skeletonization results



(a). Segmented hepatic vein (b). Skelton of the hepatic vein

Figure 48 -- cont. Hepatic vein segmentation and skeletonization results

CHAPTER V

SUMMARY AND CONCLUSIONS

V.1 Summary of the Four Chapters

This work aims at hepatic vessel segmentation in CT liver images, which is a critical processing step for morphology and topology analysis of the vascular systems for many medical applications. The main vessel systems in the liver are the portal vein, hepatic vein, hepatic artery and bile ducts. The portal vein and hepatic vein are two major hepatic vasculatures that play an important role in liver surgical planning, so our ultimate goal is to extract these two types of veins. However, the inherent geometrical properties of the hepatic vessel trees, the imaging characteristics, and limitations of the CT images make the segmentation a challenging task.

Due to the imaging protocol, the portal vein is shown clearly in the arterial phase image, while both the portal vein and hepatic vein are enhanced simultaneously during the venous phase. To achieve accurate segmentation of both veins, the first step is to separate these two vascular systems. But before separation, some other pre-processing steps are also essential.

In Chapter II, image registration is performed first to remove the subtle displacement between the two imaging phases. A rigid transformation, followed by a non-rigid registration, is used to align the portal vein in both images. Both the rigid and non-rigid registration algorithms use a normalized mutual information-based approach. The non-rigid registration algorithm is an intensity-based adaptive bases algorithm which

models the deformation field with compactly supported radial basis functions in multiple scales and resolutions. To achieve a better alignment between vessels instead of matching relatively large structures in the images, we compute the joint histogram, and thus the normalized mutual information, within the estimated range of vessel intensities. Such intensity restriction on the registration leads to a very good match between thin vessels, and reduces computation time as well.

To denoise the image, more specifically, to preserve and enhance the vascular structures while smoothing the liver parenchyma, a Hessian filter is employed in our approach. This tubular structure detection filter is applied at different scale and the width of the vessel is estimated on the basis of the scale that maximizes the filter response. To avoid the border effect of the liver, the image is normalized to the [0, 1] interval first. We have developed an automatic method to select the best value for the parameter g_{low} of the intensity transformation, which is roughly the lower bound of the intensity value for the vessels. This selection is based on the analysis of the cumulative probability function of the Hessian enhanced image and its derivative. The intensity normalization reduces the big intensity difference between the liver tissue and the background, which, in turn, reduces the response of the filter around the borders of the liver, thus decreases the interference on the boundary. Results show that Hessian filter enhanced the vessels while suppressing noise and other non-tubular structures.

In Chapter III, we proposed a vessel separation method with automatic thresholds selection based on the 2D intensity histogram analysis. It involves the selection of the two initial thresholds for the two types of veins – the portal vein and the hepatic vein voxels – by the entropy and eigen-analysis of the two Gaussian-shaped histograms, and

an iterative classifier estimation process, in which a Bayesian classifier is used to separate the two histogram clusters to segment the vessels. Results show that this approach can separate the portal vein and the hepatic vein in the venous phase image, making the following segmentation easier. It fills in the gaps between disconnected vessels and produces better results than the method that simply subtracts the portal vein of the arterial phase from the venous phase image.

In Chapter IV, a three-step iterative segmentation method was developed to extract the hepatic vessels. It involves a traditional region growing method, an adaptive directional region growing method, and a reconnection step. The traditional region growing procedure and the global skeletonization create the initial vessel and skeleton guidance for the following steps; the adaptive directional region growing algorithm is confined to a local region of interest, which adapts its size, position and orientation to the predicted properties of the vessel branches; and reconnection completes the vessel systems with disconnected pieces that have not been found in the first two steps. Step 2 and step 3 are iteratively repeated until no more new vessel branch can be found. Different schemes for the portal vein segmentation have been suggested and compared. Results show that the one combining the portal vein information in the two image phases by summation performs the best, since it preserves the exclusive PV information in each phase. Results have shown that our method produces more detailed vessel branches and recovers more branches than a standard region growing technique method both for the portal and hepatic veins.

V.2 Future Work

The approach proposed in Chapter III separates the portal vein and the hepatic vein reliably for the datasets which contain one vessel in the arterial phase and two vessels in the venous phase. In such datasets, subtraction is used to get the initial PV/HV mask. We also have seen some datasets that capture both veins in both phases. To obtain satisfactory results for these datasets, it will be necessary to develop techniques other than subtraction to acquire the initial PV/HV masks. Once the initial masks are obtained, the following separating steps are similar.

Although the methods described in Chapter II and III are fully automatic, the method described in Chapter IV has not reached this point yet. The robustness of the method can be improved in several aspects. Firstly, the intensity threshold for the segmentation currently needs to be adjusted due to contrast differences between volumes. The histogram of the liver has only one peak. One possible way to make the threshold selection automatic is to use a weighted histogram algorithm (e.g., voxels with high or low edge values contribute differently to the calculation of the histogram) to produce two peaks in the intensity histogram, one for the liver parenchyma, and one for the vessels, and a valley between them. The intensity value at the valley could be used as a good threshold. Or, algorithms could be developed to reduce the contrast difference between volumes; this may alleviate the need for threshold adjustment. Secondly, the current method has not used any anatomical information on the hepatic vessels. The liver vasculature has been well studied and it is possible that a-priori information could be used to differentiate spurious vessel branches from real ones. One difficulty, however, is the inter-patient variability observed in the vasculature.

REFERENCE

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

[2] Alighieri Mazziotti, and Antonio Cavallari., Techniques in Liver Surgery, Greenwich Medical Media.

[3] http://www.livertumor.org.

[4] Adam Hoover, Valentina Kouznetsova, and Michael Goldbaum, "Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response", IEEE Transactions on Medical Imaging, vol. 19, no. 3, pp. 203-210, 2000.

[5] Do Yeon Kim, and Jong Won Park, "Connectivity-based local adaptive thresholding for carotid artery segmentation using MRA images", Image and Vision Computing, vol. 23, pp. 1277-1287, 2005.

[6] Michael H.F.Wilkinson, Tsjipke Wijbenga, Gijs de Vries, and Michel A. Westenberg, "Blood vessel segmentation using moving-window robust automatic threshold selection", Proceedings of International Conference on Image Processing, vol. 3, pp. 1093-1096, 2003.

[7] Do Yeon Kim, and Jong Won Park , "Connectivity-based local adaptive thresholding for carotid artery segmentation using MRA images", Image and Vision Computing, vol. 23, pp. 1277-1287, 2005.

[8] Qin Li, J. You, Lei Zhang, and P. Bhattacharya, "Automated retinal vessel segmentation using multiscale analysis and adaptive thresholding", IEEE Southwest Symposium on Image Analysis and Interpretation, pp. 139-143, 2006.

[9] Yoshitaka Masutani, Thomas Schiemann, and Karl Heinz Höhne, "Vascular Shape Segmentation and Structure Extraction Using a Shape-Based Region-Growing Model", Lecture Notes in Computer Science, Proceedings of the First International Conference on Medical Image Computing and Computer-Assisted, vol. 1496, pp. 1242-1249, 1998.

[10] P. J. Yim, P. L. Choyke, and R. M. Summers, "Gray-scale skeletonization of small vessels in magnetic resonance angiography", IEEE Transactions on Medical Imaging, vol. 19, pp. 568-576, 2000.

[11] Wenli Cai, Frank Dachille, Gordon J. Harris, and Hiroyuki Yoshida, "Vesselness propagation: a fast interactive vessel segmentation method", Proceedings of SPIE: Medical Imaging, vol. 6144, pp. 1343-1351, 2006.

[12] N. Passat, C. Ronse, J. Baruthio, J. P. Armspach, C. Maillot, and C. Jahn, "Atlas-based method for segmentation of cerebral vascular trees from phase-contrast magnetic resonance angiography", Proceedings of SPIE: Medical Imaging, vol. 5370, pp. 420-431, 2004.

[13] Jing Jiang, Ming Dong, and Haacke, E. M., "ARGDYP: an adaptive region growing and dynamic programming algorithm for stenosis detection in MRI", IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 2, pp. 465- 468, 2005.

[14] Jinghao Zhou, Sukmoon Chang, Dimitris Metaxas, and Leon Axel, "Vessel boundary extraction using ridge scan-conversion and the deformable model", 3rd IEEE International Symposium on Biomedical Imaging: Nano to Macro, pp. 189-192, 2006.

[15] Marcela Hernandez-Hoyos, Alfred Anwander, Maciej Orkisz, Jean-Pierre Rouxx, Philippe Douek, and Isabella E. Magnin, "A deformable vessel model with single point initialization for segmentation, quantification and visualization of blood vessels in 3D MRA", Conference on Medical Image Computing and Computer-Assisted, vol. 1935, pp. 735-745, 2000.

[16] O. Wink, W. Niessen, and M. Viergever, "Fast delineation and visualization of vessels in 3-D angiographics images," IEEE Transactions on Medical Imaging, vol. 19, pp. 337-346, 2000.

[17] Yuan-Tsung Chen, and Ming-Shi Wang, "MR cholangiography 3D biliary tree automatic reconstruction system", Computerized Medical Imaging and Graphics, vol. 28, pp. 13-20, 2004.

[18] S. Suryanarayanan, A. Gopinath, Y. Mallya, K. S. Shriram, and M. Joshi, "Automatic tracking of neuro vascular tree paths", Medical Imaging: Image Processing, vol 6144, pp. 1495-1502, 2006.

[19] S. Aylward, S. Pizer, E. Bullitt, and D. Eberl, "Intensity ridge and widths for tabular object segmentation and registration", Proceedings of the Workshop on Mathematical Methods in Biomedical Image Analysis, pp. 131–138, 1996.

[20] E. Bullitt, S. Aylward, A. Liu, J. Stone, S. Mukherji, C. Coffey, G. Gerig, and S.M. Pizer, "3d graph description of the intracerebral vasculature from segmented mra and test of accuracy by copariosn with x-ray angiograms", Information Processing in Medical Imaging, vol. 1613, pp. 308–321, 1999.

[21] Stefan Wesarg, S., Firle, E.A., "Segmentation of vessels: the corkscrew algorithm", Proceedings of SPIE: Medical Imaging, vol. 5370, pp. 1609-1620, 2004.

[22] A. F. Frangi, W. J. Niessen, R. M. Hoogeveen, T. Van Walsum, and M. A. Viergever, "Model-based quantitation of 3-D magnetic resonance angiographic images", IEEE Transactions on Medical Imaging, vol. 18, pp. 946-956, 1999.

[23] Jian Chen, and A. A. Amini, "Quantifying 3-D vascular structures in MRA images using hybrid PDE and geometric deformable models", IEEE Transactions on Medical Imaging, vol. 23, pp. 1251-1262, 2004.

[24] Marcel Jackowski, Xenophon Papademetris, Lawrence W. Dobrucki, Albert J. Sinusas, and Lawrence H. Staib, "Characterizing vascular connectivity from microCT images", Medical Image Computing and Computer-Assisted Intervention, vol. 3750, pp. 701-708, 2005.

[25] Liana M. Lorigo, Oliver Faugeras, W. Eric L. Grimson, Renaud Keriven, Ron Kikinis, and Carl-Fredrik Westin, "Co-dimension 2 geodesic active contours for MRA segmentation", Proceedings of the 16th International Conference on Information Processing in Medical Imaging, vol. 1613, pp. 126-139, 1999.

[26] Rashindra Manniesing, Wiro Niessen, "Local speed functions in level set based vessel segmentation", Medical Image Computing and Computer-Assisted Intervention, vol. 3216, pp. 475-482, 2004.

[27] Rashindra Manniesing, Birgitta Velthuis, Maarten van Leeuwen, and Wiro Niessen, "Skletonization for re-initialization in level set based vascular tree segmentation", Proceedings of SPIE: Medical Imaging, vol. 5730, pp. 506-514, 2004.

[28] Rashindra Manniesing, and Wiro Niessen, "Shape constrained vessel centerline extraction by integrating surface evolution and topology analysis", IEEE International Symposium on Biomedical Imaging: Nano to Macro, pp. 165-168, 2006.

[29] C. M. Van Bemmel, L. J. Spreeuwers, M. A. Viergever, and W. J. Niessen, "Level-set-based artery-vein separation in blood pool agent CE-MR angiograms", IEEE Transactions on Medical Imaging, vol. 22, pp. 1224-1234, 2003.

[30] C. M. Van Bemmel, and Wiro J. Niessen, "Semi-automatic segmentation and quantification of the internal carotid artery from 3D contrast-enhanced MR angiograms", Proceedings of SPIE: Medical Imaging, vol. 5370, pp. 934-943, 2004.

[31] F. Quek, and C. Kirbas, "Vessel extraction in medical images by wave propagation and traceback", IEEE Transactions on Medical Imaging, vol. 20, pp. 117-131, 2001.

[32] T. Deschamps, and L.D.Cohen, "Fast extraction of tubular and tree 3D surfaces with front propagation methods", Proceedings of the 16th International Conference on Pattern Recognition, vol. 1, pp. 731-734, 2002.

[33] J. Tschirren, E. A. Hoffman, G. McLennan, and M. Sonka, "Intrathoracic airway trees: segmentation and airway morphology analysis from low-dose CT scans", IEEE Transactions on Medical Imaging, vol. 24, pp. 1529-1539, 2005.

[34] Cemil Kirbas, and Francis Quek, “A review of vessel extraction techniques and algorithms”, ACM Computing Surveys (CSUR), vol. 36, pp. 81-121, 2004.

[35] J. S. Suri, Kecheng Liu, L. Reden, S. Laxminarayan, “A review on MR vascular image processing: skeleton versus nonskeleton approaches: part II”, IEEE Transactions on Information Technology in Biomedicine, vol. 6, no. 4, pp. 338-350, 2002.

[36] L. Soler, H. Delingette, G. Malandain, and et al., “Fully automatic anatomical, pathological and functional segmentation from CT scans for hepatic surgery”, Proceedings of SPIE: Medical Imaging, vol. 3979, pp. 246-255, 2000.

[37] Gerald Glombitza, Wolfram Lamade, Athanasios M. Demiris, Marc-Roger Go ̈pfert, Achim Mayer, Malte L. Bahner, Hans-Peter Meinzer, Go ̈te Richter, Thomas Lehnert, Christian Herfarth, “Virtual planning of liver resections: image processing, visualization and volumetric evaluation”, International Journal of Medical Informatics, vol. 53, no. 2, pp. 225-237, 1999.

[38] Takeshi Saitoh, Masatoshi Ikeda, Kimiya Aoki, Toyohisa Kaneko, and Ryuzo Sekiguchi, “Optimal threshold for hepatic blood vessels based on structural analysis and cancer detection”, Systems and Computers in Japan, vol. 36, pp. 1-12, 2005.

[39] Noriko Inaoka, Hideo Suzuki, and Morimichi Fukuda, “Hepatic blood vessel recognition using anatomical knowledge”, Proceedings of SPIE: Medical Imaging, vol. 1652, pp. 509-513, 1992.

[40] Y. Masutani, Y. Yamauchi, M. Suzuki, Y. Ohta, T. Dohi, M. Tsuzuki, and D. Hashimoto, “Development of interactive vessel modeling system for hepatic vasculature from MR images”, Medical and Biological Engineering and Computing, vol. 33, no. 1, pp. 97-101, 1995.

[41] Catalin Fetita, Olivier Lucidarme, and Francoise Preteux, “Automated 3D vascular segmentation in CT hepatic venography”, Proceedings of SPIE: Mathematical Methods in Pattern and Image Analysis, vol. 5916, pp. 114-125, 2005.

[42] Yi Shen, Boliang Wang, Ying Ju, Jiezhen Xie, and Xiaoyang Huang, “Interaction techniques for the exploration of hepatic vessel structure”, Proceedings of the 27th Annual International Conference of the Engineering in Medicine and Biology Society, pp. 2902-2905, 2005.

[43] Yan Yang, Stephanie George, Diego R. Martin, Allen R. Tannenbaum, and Don P. Giddens, “3D modeling of patient-specific geometries of portal veins using MR images”, Proceedings of the 28th Annual International Conference of the Engineering in Medicine and Biology Society, pp.5290-5293, 2006.

[44] Paul F. Hemler, Evan S. McCreedy, Ruida Cheng, Brad Wood, and Matthew J. McAuliffe, “Vasculature segmentation for radio frequency ablation of non-

resectable hepatic tumors", Proceedings of SPIE: Medical Imaging, vol. 6144, pp.1134-1141, 2006.

[45] T. Pock, C. Janko, R. Beichel, and H. Bischof., "Multiscale medialness for robust segmentation of 3D tubular structures", 10th Computer Vision Winter Workshop, pp. 93-102, 2005.

[46] Zahlten C, Jürgens H, Evertsz CJ, Leppek R, Peitgen HO, and Klose KJ, "Portal vein reconstruction based on topology", European journal of radiology, vol. 19, no. 19, pp. 96-100, 1995.

[47] Petr Dokladal, Christophe Lohou, Laurent Perroton, and Gilles Bertrand, "Liver blood vessels extraction by a 3-D topological approach", Lecture Notes in Computer Science, Proceedings of the Second International Conference on Medical Image Computing and Computer-Assisted Intervention, vol. 1679, pp. 98-105, 1999.

[48] D. Selle, B. Preim, A. Schenk, and H.-O. Peitgen, "Analysis of vasculature for liver surgical planning", IEEE Transactions on Medical Imaging, vol. 21, pp. 1344-1357, 2002.

[49] Horst K. Hahn, Bernhard Preim, Dirk Selle, Heinz-Otto Peitgen, "Visualization and interaction techniques for the exploration of vascular structures", Proceedings of the Conference on Visualization, pp. 395-578, 2001.

[50] S.-Y. Wan, and W. E. Higgins, "Symmetric region growing", IEEE Transactions on Image Processing, vol. 12, pp. 1007-101, 2003.

[51] Shu-Yen Wan, Atilla P. Kiraly, Erik L. Ritman, and William E. Higgins, "Extraction of the hepatic vasculature in rats using 3D Micro-CT images", IEEE Transactions on Medical Imaging, vol. 19, no. 9, pp.964-971, 2000.

[52] Reinhard Beichel, Thomas Pock, Christian Janko, Roman B. Zotter, Bernhard Reitinger, Alexander Bornik, Kalman Palagyi, Erich Sorantin, Georg Werkgartner, Horst Bischof, and Milan Sonka, "Liver segment approximation in CT data for surgical resection planning", Proceedings of SPIE: Medical Imaging, vol. 5370, pp.1435-1446, 2004.

[53] K. Palágyi, J. Tschirren, and M. Sonka, "Quantitative analysis of intrathoracic airway trees: methods and validation", Lecture Notes in Computer Science, Information Processing in Medical Imaging, vol. 2732, pp. 222-233, 2003.

[54] Marius Erdt, Matthias Raspe, and Michael Suehling, "Automatic hepatic vessel segmentation using graphics hardware", Lecture Notes in Computer Science, Medical Imaging and Augmented Reality, vol. 5128, pp. 403-412, 2008.

[55] Jens N. Kaftan, Hüseyin Tek, and Til Aach, "A two-stage approach for fully automatic segmentation of venous vascular structures in liver CT images", Proceedings of SPIE: Medical Imaging, vol. 7259, pp. 725911-725911-12, 2009.

[56] M. Freiman, L. Joskowicz, and J. Sosna, "A variational method for vessels segmentation: algorithm and application to liver vessels visualization", Proceedings of SPIE: Medical Imaging, vol. 7261, pp. 72610H-72610H-8, 2009.

[57] Jaeyoun Yi, and Jong Beom Ra,"Vascular segmentation algorithm using locally adaptive region growing based on centerline estimation", Proceedings of SPIE: Medical Imaging, vol. 4322, pp. 1329-1336, 2001.

[58] C. M. Van Bemmel, L. J. Spreeuwers, M. A. Viergever, and W. J. Niessen, "Level-set-based artery-vein separation in blood pool agent CE-MR angiograms", IEEE Transactions on Medical Imaging, vol. 22, pp. 1224-1234, 2003.

[59] Wiro Niessen, Alexander Montauban van Swijndregt, Bernard Elsman, Onno Wink, Max Viergever, and Willem Mali, "Enhanced artery visualization in vlood pool MRA: results in the peripheral vasculature", Lecture Notes In Computer Science, Proceedings of the 16th International Conference on Information Processing in Medical Imaging, vol. 1613, pp. 340-345, 1999.

[60] Tianhu Lei, J.K. Udupa,. P.K. Saha, and D. Odhner, "Artery-vein separation via MRA-an image processing approach", IEEE Transactions on Medical Imaging, vol. 20, no. 8, pp. 689-703, 2001.

[61] Robert M. Stefancik, and Milan Sonka, "Highly automated segmentation of arterial and venous trees from three-dimensional magnetic resonance angiography (MRA)", The International Journal of Cardiovascular Imaging, vol. 17, no. 1, pp. 37-47, 2001.

[62] Michael Bock, Stefan O. Schoenberg, Frank Floemer, and Lothar R. Schad, "Separation of arteries and veins in 3D MR angiography using correlation analysis", Magnetic Resonance in Medicine, vol. 43, no. 3**,** pp. 481-487, 2000.

[63] Xavier Tizon, and Örjan Smedby, "Segmentation with gray-scale connectedness can separate arteries and veins in MRA", Journal of Magnetic Resonance Imaging, vol. 15, no. 4, pp. 438-445, 2002.

[64] Zhujiang Cao, "Segmentation of medical images using level set based methods", Ph.D. dissertation, 2004.

[65] Gustavo K. Rohde, Akram Aldroubi, and Benoit M. Dawant, "The adaptive bases algorithm for intensity-based nonrigid image registration", IEEE Transaction on Medical Imaging, vol. 22, no. 11, pp. 1470-1479, 2003.

[66] Alejandro F. Frangi, Wiro J. Niessen, Koen L. Vincken, Max A. Viergever, "Multiscale vessel enhancement filtering", Lecture Notes in Computer Science,

Medical Image Computing and Computer-Assisted Interventation, vol. 1496, pp. 130-137, 1998.

[67] C. Lorenz, I.-C. Carlsen, Thorsten M. Buzug, Carola Fassnacht, and Jürgen Weese, "Multi-scale line segmentation with automatic estimation of width, contrast and tangential direction in 2D and 3D medical images", Lecture Notes in Computer Science, Proceedings of the First Joint Conference on Computer Vision, Virtual Reality and Robotics in Medicine and Medial Robotics and Computer-Assisted Surgery, vol. 1205, pp. 233-242, 1997.

[68] Y. Sato, S. Nakajima, N. Shiraga, H. Atsumi, S. Yoshida, T. Koller, G. Gerig, and R. Kikinis, "Three-dimensional multi-scale line filter for segmentation and visualization of curvilinear structures in medical images", Medical Image Analysis, vol. 2, pp. 143-68, 1998.

[69] Kaleem Siddiqi, Sylvain Bouix, Allen Tannenbaum, and Steven W. Zucker, “Hamilton-jacobi skeletons”, International Journal of Computer Vision, vol. 48, no. 3, pp. 215-231, 2002

[70] Sylvain Bouix, Kaleem Siddiqi, and Allen Tannenbaum, “Flux driven automatic centerline extraction”, Medical Image Analysis, vol. 9, pp. 179-295, 2005.

[71] J. M. Fitzpatrick, D. L. G. Hill, and C. R. Maurer Jr., Image Registration. Handbook of Medical Imaging, Vol. 2, Medical Image Processing and Analysis, M. Sonka and J. M. Fitzpatrick, Eds. Bellingham, WA: SPIE Press, pp. 452-453, 2001