Pázmány Péter Katolikus Egyetem

Információs Technológiai és Bionikai Kar

Diplomamunka

Hepatic vascular network segmentation for liver surgery planning

Benis Olivér

Info-bionika mérnöki MSc

2021

Témavezető:

Dr. Benedek Csaba

Pázmány Péter Catholic University
Faculty of Information Technology and Bionics

1083 Budapest, Práter utca 50/a www.itk.ppke.hu

Thesis Proposal Form

Ikt.: I/1307/2021

Name: **Benis Olivér**	Neptun ID: **ULOKCM**
Study program: **Info-Bionics Engineering MSc (IMNM-AIB)**	

Supervisor

Name: **Dr. Benedek Csaba**

Thesis

Title: Hepatic vascular network segmentation for liver surgery planning

Summary of the thesis:
The goal of the thesis work is to develop a computer-based method for blood vessel segmentation in the liver on computed tomography (CT) images and to divide the organ into subparts based on this result. One of the main roles of the liver is the detoxification of the blood which requires a dense vascular network. A part of the liver can be removed in case of transplantation and tumors are also needed to be cut out. During these kind of surgeries it is important to consider the location of the vessels to maintain proper circulation of the remaining organ and to avoid unwanted hemorrhage. If we are able to locate the vascular network of the liver safer liver surgeries can be planned. Contrast enhanced computed tomography images are suitable to identify the blood vessel of the organ on them but doing this manually is time consuming, hard work and prone to errors. After accomplishing the tasks of the thesis work the developed segmentation algorithm will be able to perform this task automatically. As the human body contains other tree-like structures like the vascular network of the lungs it is beneficial to evaluate the segmentation algorithm also on these. Another goal is dividing the liver onto segments based on the branches of the portal vein. The resulting segments can be used to identify the boundaries of the region in the liver that can be removed during surgeries without affecting the blood flow of the remaining part.

Detailed task description:
•Review the vessel segmentation algorithms in the scientific literature.
•Based on the review develop a vascular network segmentation method and compare its results to manually segmented blood vessels done by experts.
•Investigate the applicability of the algorithm on the segmentation of different tree-like structures like the blood vessels of the lungs.
•Create a method that is able to divide the liver onto segments using the results of the vessel segmentation.
•Analyze the incidentally occurring errors of the algorithms, develop solutions to eliminate these or decrease their significance.
•Evaluate the methods based on their speed, robustness and the precision of the segmentation.
•Describe the practical applications of the work, specify the possible future improvements.

Date: Budapest, 01 September 2021

Hereby I apply for the approval of the topic of my thesis.

Signed in the Space system
01 September 2021 10:22:02

Benis Olivér
Student

Hereby I undertake to supervise the thesis work of the student.

Signed in the Space system
01 September 2021 10:36:18

Dr. Benedek Csaba
Supervisor

The topic of the thesis has been approved by the Faculty of Information Technology and Bionics.

Signed in the Space system
10 September 2021 11:36:42

Iván Kristóf
Dean

The student has regularly attended consultation sessions, and has met all the requirements of the task description. The thesis work meets the formal and content requirements described by the Annex 1 of the Education and Exam Policy (EEP).

Signed in the Space system
06 December 2021 16:00:56

Dr. Benedek Csaba
Supervisor

Alulírott Benis Olivér, a Pázmány Péter Katolikus Egyetem Információs Technológiai és Bionikai Karának hallgatója kijelentem, hogy ezt a diplomamunkát meg nem engedett segítség nélkül, saját magam készítettem, és a diplomamunkában csak a megadott forrásokat használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva / a dolgozat nyelvétől eltérő nyelvből fordítva, más forrásból átvettem, egyértelműen a forrás megadásával megjelöltem. Ezt a diplomamunkát más szakon még nem nyújtottam be.

Benis Olivér

Aláírás

Contents

1 Abstract

Locating the blood vessels in the liver is a crucial task for surgery planning. Liver resection is performed to remove tumors from the organ or for transplantation. The location of the cut is selected based on the liver segments because one of these can be surgically removed without affecting the blood flow of the others. The segments are delineated with taking the position of the vessels into account. Contrast enhanced computed tomography images are suitable to be used for vessel network segmentation. Doing this process manually is time consuming and prone to errors.

In the recent years both traditional and deep learning based methods were proposed for automatic vessel extraction. Algorithms without the need of training data can perform this task but they often require some user interaction like parameter settings and their precision is usually lag behind the best performing solutions. Learning based method can achieve better precision but they require manual vessel annotation for the training data.

During my work I have developed a new method for fully automatic liver vessel network segmentation that is based on a three dimensional marked point process (MPP) model and region growing algorithm. A synthetically generated training dataset was created consisting of artificial computed tomography images and the vascular network was annotated by the program during the generation process which means that neither real CT images nor manual segmentation was required. The model parameters can be determined fully automatically using a low number of training images of the synthetic CT scan dataset. The low number of parameters give us the possibility to modify the model if adaptation is needed for a new dataset, with possibly different CT image characteristics.

The method was tested on multiple CT image datasets both qualitatively and quantitatively and it was compared to other methods. Based on the experiments, the technique is showing promising results: while it is easily trainable it also comes near to the state-of-the-art performance. The adaptivity and robustness of the proposed 3D vessel network extraction algorithm was also demonstrated with testing its applicability for vessel segmentation in the lungs.

As the liver resection surgery planning depends directly on the precisely delineated liver segments I have developed a method that is able to locate these segments using the liver vascular network data. Skeleton images of the segmentation results are calculated and a graph is generated to represent the whole hepatic vascular network. Only minimal human interaction is needed for the algorithm to perform its task and after selecting the subtree of the vessel network for each segment, the liver voxels are classified into one of the segments.

2 Introduction

The liver has outstanding recovery capabilities but if it has been damaged irreversibly the only long term treatment is transplantation with resection surgery. Tumor removal is a similar process performed often. As the main functions of the liver are the detoxification and storage of the blood it is interlaced with a dense vascular network. The aforementioned surgeries can only be performed safely if the location of the vessels is known and this way the remaining liver part will stay functionally intact. To help the surgery planning models for the segmental division of the organ were created where each segment has its individual blood supply. This fact means that if one segment is removed the rest will remain functional. Locating the blood vessels is a requirement also for finding these segments.

On contrast enhanced computed tomography (CT) images we are able to locate the voxels of the liver which represent blood vessels relatively easily but selecting these voxels manually with high precision is a hard and slow process due to complex shape and variable size of the vascular network. Computer based automatic vessel segmentation is a possible solution for these problems and since the liver segments are even more tedious to be segmented manually performing the whole task algorithmically would help liver surgery planning.

In this thesis I collect first the anatomical and physiological properties of the liver and also the features of the organ on CT images. After that I review the scientific literature that is related to the segmentation of blood vessels and similar structures especially those that work on the liver and determine that how well the state-of-the-art methods can perform and what are the pitfalls and possible solutions of the process.

After the description of the developed segmentation algorithm and process I provide the results that was acquired by comparing the output of the segmenter to manual segmentation of the blood vessels done by experts. As the segmentation result can be presented as three dimensional image I visualize it as 3D models for easier qualitative evaluation.

I also review the literature about the liver segment models, especially the Couinaud

model and its proposed modifications. I develop a segmental classifier method that uses the three dimensional vascular network image extracted in the previous step. Similarly to the segmentation I generate three dimensional models of the liver with segment annotation based on my results included in them.

The vessel segmenter and the segmental division together from a CT image can give crucial information for planning safe liver resection surgeries fast. I evaluate the whole process and its result in this aspect and propose possible future modifications that can improve its precision.

3 Background

3.1 Computed tomography

Computed tomography (CT) is a medical imaging technique which produces a 3D image of a body region. The computer tomograph measures the radiodensity of the tissues. This density is expressed using the Hounsfield-scale which is relative, it is calculated with the comparison to the radiodensity of the water and vacuum. Water is 0 and vacuum is -1000 in Hounsfield units (HU), the bigger HU value represents larger density. In the human body the lungs has one of the smallest HU values, usually under -500 HU and the bones has the highest, it can be over 2000 HU. Most of the organs like the kidney, liver, muscles, brain have a density of about 100 HU. The CT images have relatively high contrast which can be further increased with intravenous contrast enhancing material which can reach almost every part of the body after a time.

3.2 Properties of the studied organs

The main focus during my work was the precise automatical segmentation of the liver vessels but I have also tested the adaptivity of the algorithm on vessel extraction from lung CT images. The blood vessels of the two organs differ in multiple aspects: the intensity contrast in comparison with the non-vessel voxels, the diameter and the branch structure. These differences make it possible to test the proposed segmentation method more extensively.

3.2.1 Liver

The liver is an organ in the abdominal cavity located directly under the diaphragm which weights about 1500 g. It consists of a smaller and a larger lobe. It has multiple functions: acts as glycogen storage, synthesizes and transforms lipids, breaks down amino acids, produces proteins, detoxificates and stores blood and as an accessory digestive organ it produces the bile [1].

It is located mainly on the right side of the body but its left lobe is partly on the other side of the spine. The surface is divided onto two parts: the upper, convex is the facies diaphragmatica, the lower is the facies visceralis [1].

The porta hepatis is the gate where the blood vessels are get into the liver. Also, the bile ducts leave the organ here leading the bile into the gall bladder. There are two input sources of blood of the liver. The hepatic artery is responsible for about the quarter the blood supply of the organ. The rest of the blood reaches the liver through the portal vein. This blood vessel has a larger diameter has than the artery. The blood leaves the liver through the hepatic veins.

The blood vessels divide into branches and based on the location of the secondary branches we can locate the liver segments. These segments are not visible on the organ, there is not any morphological difference or separation between them, they are only determined based on the vascular network. Surgeries consider the location of the segments because one can be removed without affecting the blood flow of the others [1].

Multiple nomenclature were proposed for the division of the liver and these can even differ in the number of the segments. In 1957 Couinaud described an 8 segment model [2] which is based on the portal vein and hepatic vein branches and it is mainly prevalent in European countries. The 6 segment model [3] proposed by Goldsmith and Woodburne can be deducted from the location of the hepatic veins. Later, in 1982 Bismuth [4] combined these two nomenclature resulting in 8 segments where one is divided onto two subsegments. With more recent advancements even subsegmental division for surgery planning was made possible. The differences of the vascular anatomy of the liver can cause high variance in the number of the segments located in the liver [5].

Both the portal vein and the hepatic veins can have anatomical modifications relatively often which do not hinder the functionality of the liver but they can make the successful segmental division harder.

Computed tomography is one of the most used imaging techniques of the liver together with ultrasound (US). In comparison with US using CT the body fat does not affect the result. The larger segments of the vascular network is visible even without contrast material. Injecting iodinated contrast material makes the whole vascular network distinguishable from the liver parenchyma, the voxels of the blood vessels have an increased intensity. In this case the contrast enhancement is detectable in the hepatic artery first and in the portal vein only later.

3.2.2 Lungs

The lungs are the main respiratory organs of the humans, located on both sides of the chest filling a large portion of its volume. Their outer boundaries are determined by the ribcage. Between the two lungs the mediastinum can be found which includes among others the heart, the great vessels, the esophagus and the trachea. As the connections with the other parts of the body are mainly running through the hilum on the medial surface, the organ is connected to multiple structures of the mediastinum. The trachea divides dichotomically into the two main bronchi and these enter the lung where they are branching multiple times similarly. The lungs are connected to the heart through vessels as part of the pulmonary circuit, these are the pulmonary veins and arteries. As the function of these vessels is only the gas exchange, the nutritional supply is carried out through vessels of the systemic circulation [1].

On computed tomography images the volume of the lungs appear mainly as low intensity voxels (around -1000 - -800 HU) caused by the air inside. The vessels appear as mainly homogeneous, relatively high intensity network but the cross section of the bronchial tree is a high intensity ring with low intensity voxels inside as this consists of hollow, air-filled tubes. The contrast of the voxel intensities between the neighboring areas and the lung and also the bronchial and vascular network and the other parts of the lungs is high.

3.3 Theoretical description of the segmentation algorithms

The developed vascular network segmentation method described later is the combination of a three dimensional marked point process model and 3D region growing algorithm.

3.3.1 Marked point process model

The basis of my vessel extraction method is a marked point process (MPP) model based segmentation. The MPP approach is used in the field of image processing among others for detecting unknown number of objects which have similar shape (e.g. tree crowns on aerial images can be represented by circles). In my model not distinct objects but segments of the vascular network connected to each other are represented by tubes.

The image space in case of three dimensional CT images is $I = [0, x_{max}] \times [0, y_{max}] \times [0, z_{max}]$. In this space there is a random configuration of points denoted by P and each point has a respective mark. The mark space M is based on the parameters of the mark's

shape. The marked point process model in this case is $X = P \times M$.

Prior knowledge consisting of geometrical constraints are assumed that can help choosing the shape of the mark.

The potential is a measure that the algorithm aims to minimize. It consists of the data term of individual objects to measure their fitness in multiple aspects and the prior term that is based on the expected relationship between objects and the global configuration.

The data term takes the features of the underlying image into consideration, i.e. it describes how well the marked point (MP) fits onto the image based on the intensities of the neighborhood.

These prior term control the interaction of the objects for example with penalizing overlap or promoting connections, similar orientation.

Optimization

Optimization is the process of minimizing the potential function for the configuration with iterative algorithms. As this function is non convex choosing a suitable algorithm for optimization have an important role in finding the best configuration.

Multiple ways have been proposed to simulate the underlying point process like the multiple birth and death (MBD) or the reversible jump markov chain monte carlo (RJMCMC) method.

Reversible jump markov chain monte carlo The RJ MCMC simulation method was described by Green in [6]. If x is the current state, the target state in one iteration can be written as x'. The type of the move is denoted by m which can be for example birth or death of an object, translation, modification of shape parameters. These modifications are performed by so called kernels. Between these states there is a probability $q_m(x, x')$ of moving from x to x'. $\pi(x)$ is the distribution of state x. The acceptance probability α_m can be in this case calculated as follows:

$$\alpha_m(x, x') = min\left\{1, \frac{\pi(x')q_m(x', x))}{\pi(x)q_m(x, x')}\right\}$$

Calculating the move probability in each case can yield high computational time.

Multiple birth and death The multiple birth and death method is a faster alternative to the RJ MCMC dynamics. In the initial state the configuration is empty. In the classical case every iteration consists of a birth and a death step.

The birth adds new objects to the configuration. The first advancement is that the number of objects in created in a single iteration is not limited to one. Also, instead of calculating the acceptance from the move probability the parameters of the objects (both the coordinates and the shape parameters) can be entirely random but to speed up the convergence prior information is incorporated into the selection of these which can be achieved using a birth map. This map incorporates some prior information about the location of the points and this way instead of uniform distribution, for different regions the probability of MP birth will differ.

After the birth of an unknown number of objects their data term is calculated and they are sorted by descending order. In the death step the algorithm considers every object in this order and calculates a death probability for each of them based on the data term. The current object is deleted with this probability and then the method steps to the next marked point.

This iteration is continued until convergence which is reached when each of the objects created at the most recent birth and none other are removed during the death step.

3.3.2 Region growing algorithm

The main idea of the region growing algorithm [7] that starting with one or more seed points the neighboring voxels are added to the already selected one if they are similar enough to the voxels of the region. This similarity is usually measured with comparing the intensities. The growing can happen on images both in two and three dimension and the type of the considered connections between the neighbors can also differ.

In case of the application of this algorithm it is important to consider its required time. If the goal is to segment a large region especially if it is 3D, the algorithm will be slow. Parallelization of this method is possible [8] and it is useful to consider this kind of implementation.

3.4 Related work

In three different subjects it is important to review the scientific literature. The overview of the marked point process based segmentation methods can be useful for the implementation of an efficient and precise model. It is also important to collect the specific characteristics of the liver CT images and how proposed methods can be adapted to have respect to these. Lastly, liver segment classification methods are similarly important to review.

3.4.1 Synthetic vessel image generation

The problem of the low number of high quality annotated medical image datasets and the work demand of the manual organ segmentation gave rise to many synthetic blood vessel network generators, sometimes even incorporating the properties of the CT images. These methods usually require parameters to be set for realistic results or use real images and manual segmentation to fine tune the generation of the image.

In [9] two different deep learning methods, variational autoencoder (VAE) and generative adversarial network (GAN) were used to generate vessel segments. Three different problems were mentioned here, generation of 2D and also 3D vessel images and creating stenosed artery models where these vessels are obstructed and this changes their shape. The creation of a whole network with branching and varying vessel diameter was not part of this paper but, especially with GAN, they could generate vessel shapes very similar to the training samples but the disadvantage of this method is that a high number of images is needed for training and as the goal here is different, the characteristics of the CT images and the liver anatomy is not taken into account.

The proposed method in [10] consists of creating a sequence of rules in the form of a string to generate the current vessel image and after that, using these instructions the vessels are generated. The rules give the possibility to generate different images stochastically. After a set of rules describing for example the length, diameter and orientation of the vessel segments are created the images themselves are generated. During this work B-spline interpolation is used to connect the nodes and on the resulting images the intensity distribution will follow the profile of real vessels on CT images.

VascuSynth [11] is an open source software which can generate blood vessel networks given real world anatomical and physiological parameters. The goal in this paper was to create a method which can be used widely to evaluate and compare segmentation methods. The parameters we need to provide are the perforation point, where the vessel network starts at, the pressure here and also at the terminal nodes, the initial flow, the number of terminal nodes, the oxygen demand map, the viscosity.

3.4.2 MPP-based methods

The papers which used marked point process or similar method are mainly focusing on two dimensional images including aerial imagery [12][13] and microscopy recordings [14]. I also found methods focusing on blood vessel extraction on two dimensional medical images [15].

The authors of [12] proposed a general marked point process model for image segmentation. A 2D mark library is created here to give the possibility of adaptation to different applications. Different line and rectangle marks are defined based on the location of their outer area. This way the method is capable to recognize even ends of a line where the outer region at one side of the shape is not needed to be considered. The properties considered for the calculation of the data term are the average voxel intensities inside the object and around it and also the standard deviation is used. This is a good compromise so that the algorithm is fast enough but also capable to evaluate the fitness of the object. In contrast to the widely used interaction properties like comparing the orientation of the objects, here only the overlap is penalized. This way the tuning of the parameters of the different terms can be avoided which is done usually with trial-and-error. Jump-diffusion is used as optimization algorithm. Both birth-and-death and switching kernel is implemented here, this means that the parameters of the existing objects is modified without changing the number of the objects. The evaluation have shown that for some applications the method cannot perform well. Taken this into account they have introduced two prior terms based on the connection and the mutual alignment. This modification has increased the computational time and also the parameter tuning problem holds but it is only needed on more complex images and can increase the performance in these cases.

The goal of the method proposed in [14] was to locate carbon fiber segments on 2D microtomography and microscope images. The shapes assigned to the marked points are ellipses. Based on prior assumptions they could limit the geometrical parameters of the generated ellipses like the length of their axes relative to each other. A novelty in this method that the gradient of the image is used when calculating the data term. MCMC simulation is used to perform the optimization. During this process besides the creation and removal of the object multiple modifications are made on them: the modification of their size, rotating and moving the ellipses.

The underlying model in [16] and [15] is similar. In case of the first one two marks are used simultaneously: ellipses and 2D tubes. The reason for this choice is that these two sorts of geometric primitives can model the 2D sections of 3D tubular structures at different angles. Two possible applications were discussed, 2D microscopy images of fiber materials and the recognition of roads and houses on aerial images. Overlap, connection and length prior is used, the latter is to promote longer marks. In [15] due to the properties of the retina fundus images only 2D tube marks were used. This paper claims unsupervised segmentation but the priors require trial-and-error tuning which,

as discussed earlier, is disadvantageous. Multiple birth-and-death algorithm is used for optimization. This method includes a birth and death kernel, a perturbation kernel to modify the parameters of the marked points and the method proposed in the first paper also uses a switch kernel to change the type of the mark between the two options. An example result is shown in Figure 3.1.

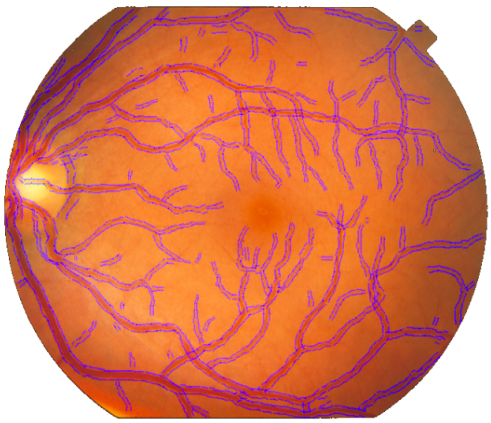

Figure 3.1: 2D connected tube MPP result on fundus image [15]

A marked point process model to be used for crowd counting is proposed in [17]. Multiple constraints were applied here to take the properties of a crowd into account. The people on these images are expected to stand or walk which means that after calculating the orientation of the image the possible orientations of the marked points are heavily limited. The shape of the marks is complex, it can fit onto a pedestrian precisely. A shape prototype is learned first and it is modified for the different marked points. In this method the optimizer is RJMCMC with birth, death and update steps for the marked points.

Point cloud based LiDAR recordings give the possibility of using 3D MPP models for different purposes. A tree detector was proposed in [18] and a archeological tumulus detector in [19]. The model in [19] contains circles as the shape of the marks and the properties along the third axis are built into the marked points as two parameters which is the base height of the points in the region and the standard deviation of it. The optimization differs in several aspects from the earlier ones. Both the parameters and the location of the MPs are modified in separate steps. Also, besides the conventional birth and death of points there is also a possibility to remove multiple objects in a neighborhood in the same interaction. In [19] the shape is a half ellipsoid which can approximate a tumulus well. The data term here is based mainly on the difference of

the underlying surface points from the ideal half ellipsoid surface. In this case multiple birth and death algorithm was used for finding the correct marked points.

3.4.3 Hepatic vascular network segmentation

Several papers have detailed methods which can be used for retinal blood vessel extraction. The data is not 3D like in case of a liver CT image but the structure of the vessels is similar which means these ideas can be helpful in case of the liver also. The input data in these cases was a set of fundus images which are two dimensional recordings of the rear of the eye with optical camera. The first algorithm [20] consists of three stages: preprocessing with two parallel processing paths, the main segmentation and postprocessing. In this they proposed an algorithm which is based on Gaussian mixture model classification, an other one [21] uses fuzzy clustering and the third one [22] is uses edge enhancement and thresholding among others to extract the vessels. All of these methods are relatively simple and easy to implement but in case of the liver they cannot perform accurately enough. The last method [15] for retinal vessel extraction was detailed in Section 3.4.2.

Published methods aiming to achieve the segmentation of the hepatic vessels on CT images automatically include both bottom up and top down segmentation algorithms.

The technique proposed in [23] divides the solution into two parts after an initial noise filtering. The parts are aiming to segment blood vessels with different thickness. Both tasks consist of two steps, first performing only an initial detection and then based on this result the segmentation itself is carried out. For thin vessels, Gaussian filter and region growing are used, the thick ones are segmented roughly with K-means clustering and this result is refined using active contour algorithm. The method shows that between the different segments of the vascular network there is a high variation in size, which fact implies that the same technique cannot always be applied effectively for all vessels.

In the work of [24] the authors use a Hessian matrix-based filter to enhance vessels. This so called vesselness filter is often used on the vasculature of other organs like lung or the retina but in case of the liver it would also enhance the contour of the organ and the parenchyma, therefore they improved the filter to avoid this phenomenon. On the enhanced images they have used a fuzzy connectedness algorithm to segment the vessels. They were able to improve the algorithm that way that it needs only a single seed which is generated automatically.

Neural network based top down methods are also effective for the task. In [25] a 3D

U-Net architecture is applied on preprocessed CT images where the loss function is based on the Dice coefficient. The neural network can be trained with incomplete labels where some blood vessel segments may be missing. The dataset was divided into training and testing samples by hand so that it yields the best result.

Another U-Net based method was proposed in [26] which directly uses the Dice coefficient in the loss function. Two important modifications were made here in the architecture. First, with their proposed attention-guided concatenation (AGC) module the method is able to select the more relevant low-level features using the high-level ones during training. Second, to improve the performance they also included in the architecture a multi-scale feature fusion block which is used to capture multiscale data instead of applying convolutions.

The method proposed in [27] is also learning-based but instead of a U-Net architecture they implemented a simpler convolutional neural network (CNN) model. The training samples are noisy, rough blood vessel labels, therefore only a small number of manually created labels are required for this method. Using sparse dictionary learning and logistic regression classifier trained on the manual vessel annotation data. These models are used to generate noisy vessel segmentation for whole CT volumes. The result acquired this way is used to train the 3D CNN. In comparison with using vessel annotation data done by hand for training the network with the extra step they could decrease the amount of manual segmentation required for the algorithm.

3.4.4 Delineation of liver segments

The liver segments are determined based on the branches of the hepatic vascular network which means that blood vessel segmentation is required beforehand for this task. Because of this fact liver segment delineation methods include a vessel segmentation step but the algorithms used for this task are often simpler than the ones proposed in the previous section because the process of finding the segments is only moderately sensitive for the quality of the vessel segmentation result.

In [28] a framework has been proposed with graphical user interface which can be used to perform the required task with the help of human interaction. First the program segments different anatomical landmarks of the liver. Multiple segmentation algorithms are included like region growing and active contours and the user can select seed points and set different parameters of the algorithms. The vessel network is also segmented similarly after this step and the quality of the result is evaluated by the user. If the quality

is poor, the user is able to modify the parameters and perform the segmentation again. The vessel network is extracted using the interactively selected portal vein root and lower and upper intensity thresholds for the vessel voxels. After the incidental manual separation of the venous and arterial network a three dimensional model is generated and presented to the user from the results. Locating the portal vein branches and from these the segments is based on another paper [29]. There is an alternative segment classificator which uses four planes to divide the liver. The planes have to be determined by the user. As it is a case for the partial results the end results are visualized in 3D in the developed program.

The method described in [30] also includes a graphical interface where human interaction is needed to select and correct the result in multiple steps of the process. The main steps are liver extraction, vessel extraction and the delineation of the liver segments. The liver is segmented with first locating multiple seed points using histogram analysis inside the liver. These seed points are used in the next step where two level-set methods are applied to find the exact liver region. The paper details the possibilities in case of low intensity tumor regions in the liver because by default these are not included in the result. In case of inner tumors simple hole filling is adequate but if the tumor is located at the side of the organ an arc has to be calculated to close the opening on the result beforehand. The vessel extraction is similar to the method which was used for the whole organ. With the analysis of the histogram of the liver voxels two peaks can be found and the peak with larger intensity usually represent the blood vessel voxels. The location of the peak is determined for the image and several histogram window widths were selected based on 30 CT images beforehand. After the histogram windowing seed points are selected on the remaining region and the same level-set methods are used for segmentation that were mentioned earlier. The user has to select the best result on the interface from the ones with different window widths. The vessel extraction result is skeletonized and the result is presented to the user, who can select the root branch points for the portal vein branches. From these points until the end of the branches the voxels are selected and the branch is extracted. Using nearest neighbor approximation each voxel of the liver region is classified to one liver segment.

A solution for liver segment extracting is also described in [31]. In this case multiple semi-automatic or manual segmentation methods are developed which can be used for the delineation of the liver. This method requires the least amount of interaction among the reviewed methods. The vessel network is segmented using three dimensional region

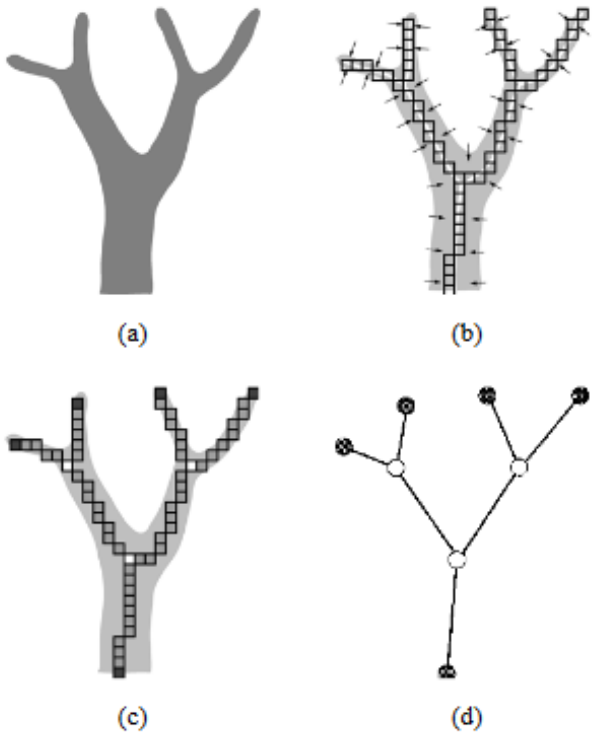

Figure 3.2: Steps of the vascular network graph generation (a) vessel segmentation result; (b) skeletonization; (c) identification of end and branch points; (d) the generated graph [31]

growing which requires one or several seed points and a threshold to limit the growing process. The seed point is located at the root of the vessel network and has to be selected manually. The threshold selection is an automatic process which is based on the assumption that at the correct threshold there is a jump in the number of the selected voxels. As the segmented voxels are stored for the whole range of threshold, it is easy to fine tune the parameter if the produced result is not satisfactory. The next step is the skeletonization of the result using 3D thinning with the help of distance transformation for symmetric erosion. After that a graph is calculated which represents the vessel network according to the detected branch point and end points. These steps are shown in Figure 3.2. The goal is the segmenation of the portal vein as the hepatic artery and the hepatic vein is not needed for the segment classification. It is possible that the artery and hepatic vein are segmented together with the portal vein, which can be eliminated with analyzing the graph: from the root of the portal vein the direction of the branches are determined. Observing the radius of the branches probable roots of different vessel networks can be found. If this is the case points where branches with opposite directions meet can be located and the vessel networks are separated here.

4 Creation of a synthetic liver CT image dataset

The parameter tuning step described earlier involves the generation of marked points on CT images and selection based on their Dice-score in comparison to ground truth data. This means that to select correct weights manual blood vessel segmentation is needed and in case of it is not precise enough it can hinder the quality of the result. Taking into consideration the characteristics of the CT images including the typical Hounsfield-values and the type and quantity of the noise and also the anatomical properties of the liver it is possible to create computer generated, synthetic CT images. In this case if first the vessel network is generated and the parenchyma and the noise is added only later, a perfect ground truth image can be acquired automatically. The properties of the objects on the images are limited to achieve realistic results but the object sizes, the branching of the vessel network and the noise are usually stochastic and using a method a whole database can be generated containing unique images.

4.1 Vessel network generation

The VascuSynth software [11] mentioned earlier is sufficient to be used for liver blood vessel network creation with additional modifications.

In [11] they applied B-spline-based spatial warp to deform the vessel segments so that they will be curved similarly to real ones. As this is not implemented in the software I have modified the source code to be able to create curvatures. My goal was to integrate the modifications so that the original image drawer can still be used.

The first part of the synthesis is the graph creation. In this step the coordinates of the nodes, their connections and the radii of the vessels are determined. This sort of data is used to draw the image where by default the nodes are connected with straight lines. This property indicates that the curvature creator should be built in between these two steps.

All the nodes except the initial one are considered. First the direction vector ($v_{parentDir}$) from the node to the parent is calculated. This vector is rotated around the z-axis using the rotation matrix with a random α angle where $0° \leq \alpha \leq 45°$ as shown in Equation 4.1. Also, this rotation is not applied for node pairs with a distance smaller than 10 voxels to avoid discontinuity of the vessel in these cases. The resulting vector ($v_{rotated}$) determines the curvature of the segment between the two nodes.

$$v_{rotated} = \begin{bmatrix} cos(\alpha) & -sin(\alpha) & 0 \\ sin(\alpha) & cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} * v_{parentDir} \tag{4.1}$$

The $v_{rotated}$ vector is rotated around the original direction vector using the Rodrigues' formula with a random β angle so that the orientation of the curvatures will be different (Equation 4.2).

$$\begin{aligned} v_{curveDir} = v_{rotated} * cos(\beta) + v_{parentDir} \times v_{rotated} * sin(\beta) + \\ + v_{parentDir} * v_{parentDir} \cdot v_{rotated} * (1 - cos(\beta)) \end{aligned} \tag{4.2}$$

Based on the $v_{curveDir}$ vector the points of the curve can be calculated between the two nodes. During the drawing of the tree each considered voxel is substituted by the closest curve point. The vascular tree drawer expects the coordinates of the nodes between which two the current voxel is. A vector is calculated from a voxel on the non-curved, original line to the corresponding voxel on the curve. The node points are shifted with these vector for every curve point. With these new nodes given to the tree drawer it can generate a curve without modifications.

The most important parameters are the initial and the terminal blood pressures, which according to [32] are about 120 mmHg and 6.5 mmHg for the arterial and 7.5 mmHg and 6.5 mmHg for the venous network, respectively. The two networks indicate that the generation should be implemented in two steps. As proposed in [33] generating one of the networks we can create an oxygen demand map for the other one to avoid overlap of the vessels. First the arterial network is generated, and using this piece of data an oxygen demand map is calculated where the demand of the regions of the vessels are set to 0 and in case of the rest it is 1. This way during the generation of the veins the VascuSynth software selects the path so that overlap with the arteries will be avoided.

The two networks are combined onto a single 3D image and the binary ground truth is extracted using thresholding. After that noise is added to imitate the ones on the real

CT images caused by the imaging technique and the properties of the organ. Instead of the original CT recordings, the goal is to create similar ones to the preprocessed images because it is easier to describe their noise and intensity characteristics. The intensity of the non-vessel voxels is a random value between 0 and 255. After that a Gaussian white noise with 0.01 variance is added to the image.

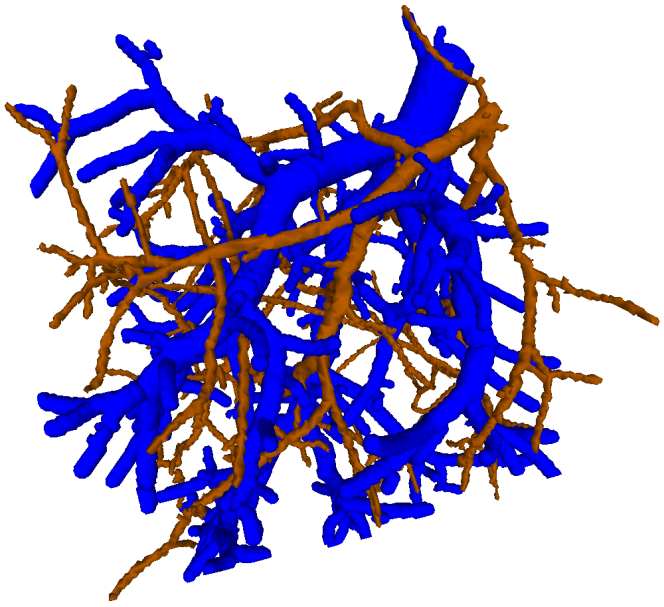

Figure 4.1: Model of a generated liver vessel network image with arteries (brown) and veins (blue)

4.2 Results evaluation

The VascuSynth software were proven to be a good selection from the scientific literature for my goals because it is very well documented which means it is easy to use and also to modify. Another advantage was the physical and physiological background that was taken into account during the development of the software. This fact gives the possibility to imitate the diameter variance and branching of the real blood vessels really good.

The images are needed to be evaluated before it can be used for parameter tuning. In case of a large histogram difference between the synthesized and real CT images the selected weights will not be useful in finding correct marked points during the segmentation on the real images.

The first evaluation method was the comparison of the histograms between the real and the synthesized images. The distance metric was the Euclidean distance. Before the

	Avg. Euclidean distance
Between synthesized images	$8.049 * 10^{-4}$
Between real images	0.01847
Between synthesized and real images	0.01950

Table 4.1: Comparison of the histograms of liver regions on CT volumes

comparison of the different kinds of images I measured the typical difference between images having the same type. As the size of the images are different the histograms were normalized first. Three real CT images were compared to three synthesized ones and also to each other. In the table the average distances can be seen for each cases. The results of the comparisons are shown in Table 4.1.

Based on these measurements we can see that the synthesized images are very similar to each other, the difference in the other cases are higher but the average Euclidean distance of the histogram values of the synthesized and the real images are very close to what we get in case of comparing real images to each other. This small difference can be caused by the lack of tumors on the synthetic images and the inclusion of non-liver regions on the real ones if there are imperfections on the liver mask. The voxels which represent a tumor on the images have the lowest intensity inside the liver and the vessel voxels are the brightest here, the false positive marked points caused by the tumors can be easily avoided, this fact makes it possible that generating tumors is not important and that the images would be appropriate to be used in the segmenter.

I have also made comparisons on the direct neighborhood of the blood vessels. First I have selected several regions on both types of images. All these regions are on two dimensional slices and their location is determined by the bounding box of the vessel ground truth data. For both kinds of images 12 regions were selected on the CT images which were used for the first evaluation method. Same as before, during the comparison of the whole volumes images of the same type were compared to each other and also every synthetic one was compared to every real one. This comparison can give more insight how well the synthetic dataset can be applied because it evaluates the reality of the intensity distribution of the vessel itself and also the difference between the vessel voxels and its neighborhood which are the most important to find the correct location for the marked points.

As it can be seen in Table 4.2, similarly to the first evaluation the local histograms of the blood vessel regions tend to be more similar to each other on the synthetic images.

	Avg. Euclidean distance
Between synthesized images	0.1409
Between real images	0.2589
Between synthesized and real images	0.2131

Table 4.2: Comparison of histograms of local blood vessel regions

The real images have more diverse local histograms. In contrast to the whole images, in case of the local vessel regions there is even more similarity in terms of the histogram between the different types than in case of the real images. This phenomenon can be caused by the fact that by this comparison the regions farther from the blood vessels are not included in the histograms. As I mentioned earlier, avoiding the false positive cases is possible in the segmenter which means that the similarity in case of the local vessel region gives more information about the suitability of the synthetic images.

It can be seen that based on these evaluations the synthetic images are similar enough to real CT scans for my purposes. Together with the ground truth data the generated images can substitute real CT images in case of parameter tuning and even for evaluating the performance of both conventional and learning-based liver vessel segmentation algorithms. Currently 20 three dimensional synthetic images are generated but as the methods are integrated together it is possible to generate new volumes easily at any time.

5 Proposed segmentation method

The segmentation workflow consists of four steps. *First* the CT image is preprocessed. *Second* the blood vessels are extracted approximately with a 3D tube MPP model. The result of this step is corrected in the *third* step which ensures correct orientation and connections between the tubes. In the *fourth*, post processing step, based on the marked point tubes the final result is obtained with region growing algorithm. The main steps of the segmentation are shown in Figure 5.1.

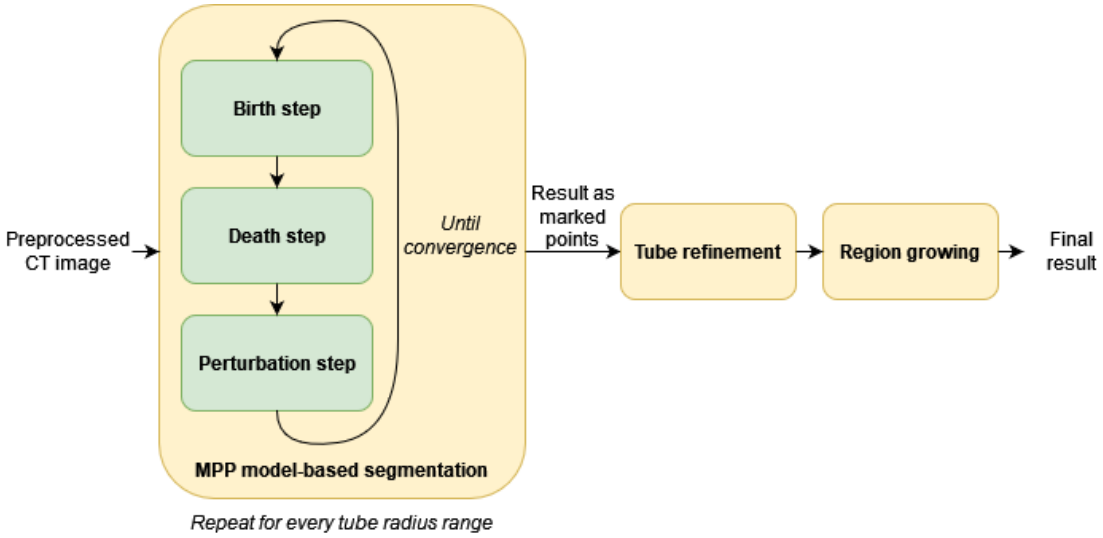

Figure 5.1: The main steps of the vascular network segmentation

5.1 Preprocessing

The input is a contrast enhanced CT image and the liver mask which tells us the location of the liver voxels. The intensities of the non-liver voxels are set to the same value. The mean intensity is calculated for the liver voxels. Next we aim to set that the root mean squared (RMS) contrast to be similar for every liver. This property can be achieved with windowing the histogram where the center of the window is the mean intensity and the width is based on the standard deviation of the voxel intensities. The

slices are stored as 8 bit images so the intensity values are modified that way that the minimum is zero and the maximum is 255.

The images are scaled to the half of their original size because the information is still sufficient for the extraction and the computational time can be greatly reduced.

5.2 Marked point process model

The models proposed in [16] and [15] use ellipses and rectangles as representations of tubes in 2D. Instead of using that 2D approach, our model works with 3D cylinders to cover blood vessel segments. The MPP potential function is built up from the data term and the overlap and connection potentials. The optimization is based on the multiple birth and death algorithm [19], which is applied multiple times while decreasing tube radius ranges.

5.2.1 3D tube mark

A marked point describing a cylinder object is shown in Figure 5.2. The position of the object is defined by the three spatial coordinates of the cylinder's center point. In addition, the object has four geometric parameters (i.e. markers): its height, the radius and two angles which determine the orientation in 3D. For the height and radius parameters there is a range of possible values which is based on the properties of the blood vessels on CT images. The space of the markers is the following: $M = [h_{min}, h_{max}] \times [r_{min}, r_{max}] \times [0, \pi] \times [0, \pi]$, here h_{min} is the minimum, h_{max} is the maximum height, r_{min} is the minimum and r_{max} is the maximum radius value.

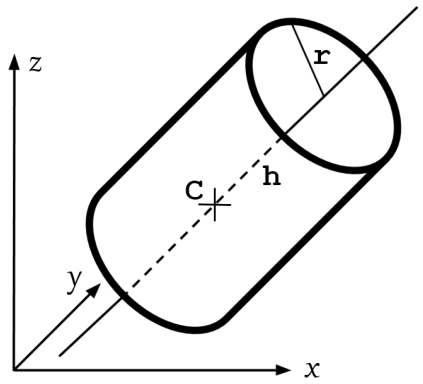

Figure 5.2: Cylinder which was used as mark

5.2.2 Potentials

We need to evaluate how well the cylindrical objects created during the extraction fits on the image as a blood vessel segment. We calculate a potential for every cylinder which can be obtained as a sum of multiple subterms: the object potential (v_{obj}) which is based on the intensities in and around the cylinder, the overlap potential (v_{ol}) that is calculated for every pair of objects and the connection potential (v_{conn}) which is based on the connection between two objects at their ends. The whole expression that we need to minimize for object a is defined by Equation 5.1 where η, θ and ξ are weights of the terms.

$$v_a = \eta v_{obj_a} + \theta \sum_{\substack{i=0 \\ i \neq a}}^{n} v_{ol_{ai}} + \xi \sum_{\substack{i=0 \\ i \neq a}}^{n} v_{conn_{ai}} \tag{5.1}$$

Object potential

On the contrast enhanced CT images the blood vessels have higher intensity voxels than their surrounding. The goal is to ensure that the inner average intensity is higher than the mean intensity outside the cylinder, but it should be similar with the average at the bases because both regions represent blood vessel parts.

The object potential describes how well the object fits onto the image based on voxel intensities. Three regions are compared to determine this potential: the inner region, the out region and the end region. The inner one is the cylinder itself, the out region is enclosed by the object and a cylinder with the same height but two voxels larger radius, the end region consists of two cylinders at the bases of the object with the radius of the object and two voxels height.

Equation 5.2 shows the object potential where $mean_{in}$, $mean_{out}$ and $mean_{end}$ are the inner, out and end mean intensity respectively, dev_{in} is the standard deviation of the inner region, $maxInt$ is the largest possible voxel intensity and α, β, δ and γ are coefficients.

$$v_{obj} = \frac{\alpha|mean_{in} - mean_{end}| + \beta|maxInt - mean_{in}| + \delta dev_{in}}{\gamma(mean_{in} - mean_{out})^2} \tag{5.2}$$

Overlap potential

The overlap between the cylinders should be avoided, which constraint can be ensured with the overlap potential v_{ol}. This term is calculated as in [16] from the overlap ratio

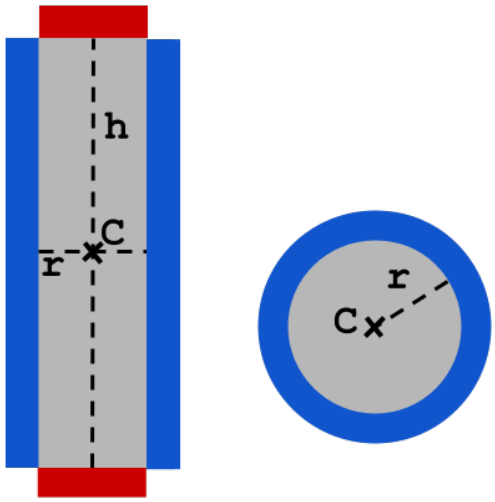

Figure 5.3: The inner (grey), out (blue) and end (red) regions on the longitudinal (left) and cross section (right) of a tube

of the cylinders and if it is over a threshold, the marked point will be deleted with $p = 1$ probability:

$$R = \frac{V_{tube_i} \cap V_{tube_j}}{min(V_{tube_i}, V_{tube_j})}$$

$$v_{ol} = \begin{cases} R, if R < T_{ol} \\ \infty, otherwise \end{cases} \tag{5.3}$$

Connection potential

To cover the whole region of the blood vessels with the cylinders we need many of them connected together at their ends. This way the segments can follow the curvatures and branches. The connections are promoted with the connection potential. With extending the connection circles in [16] to 3D we create spheres at both ends of each cylinder. The overlap between spheres of two cylinders means that they are connected, and the higher the ratio of overlap the stronger the connection. For a single marked point the connection potential contains the overlap ratios of both of its spheres with all the other marked points. Since the connected tubes should have similar thickness there is a term included in the potential v_{conn} which penalizes large radius differences:

$$R_{ij} = \frac{V_{sph_i} \cap V_{sph_j}}{min(V_{sph_i}, V_{sph_j})}$$

$$v_{conn} = -\varepsilon(\sqrt[3]{R_{Fij}} + \sqrt[3]{R_{Bik}}) + \zeta(|r_i - r_j| + |r_i - r_k|) \tag{5.4}$$

Here ε and ζ are the weights of the two terms.

5.3 Parameter tuning

The weights of the different potential terms influence the result greatly because they determine the importance of the different energy components relatively to each other. The values of the weights could be determined by trial and error but this choice can be time consuming and it is easy to miss the best setting for our purposes.

The data term parameters, including the α, β, δ and γ coefficients and the η weight parameter are determined automatically. Since the remaining parameters of the model do not depend on the image data, they are selected beforehand and are fixed.

For the data term's parameter tuning, five CT volumes were used from the 3Dircadb dataset along with their ground truth data. First, cylinder objects are generated randomly. Because of the high variation in the size of the blood vessels the weights should be different for larger and smaller vessels, respectively. This requirement is fulfilled by a multilevel operation, which we discuss later. The tuning is done for each level separately, which yields that the accepted size range of the generated cylinders are based on the current level.

The Sørensen–Dice coefficient (SDC) is calculated for each generated marked point but since with this measure the whole vascular network cannot be compared to a single cylinder, only a part of the ground truth data derived from the neighborhood of the cylinder is considered. However, uncontrolled MP generation may result in almost entirely low precision cylinders, spoiling the training process that may result in poor weights. To avoid this situation the possible range $[0, 1]$ is divided into five equal subranges and 25 marked points are selected with Dice coefficients in each of these ranges. This way a more uniform distribution of SDC values can be achieved.

For one level, 625 marked points are generated alltogether, which are used to determine the weights that give the best result. The $mean_{in}$, $mean_{out}$, $mean_{end}$ and dev_{in} values are calculated and the possible weight distributions are searched exhaustively: with every set of weights the object potentials are calculated and analysed. The goal is to ensure

that the possible largest number of cylinders with high SDC values have low potentials, while at the same time the ones with lower coefficients have large potentials. SDC over 0.8 was considered high here.

5.4 Optimization

Finding the correct configuration of cylinder objects is an iterative process. In every iteration first birth and death steps are executed to create points stochastically and remove the ones which cannot be accepted based on their potentials. Next, in the perturbation step the parameters of the marks are changed to achieve a better fit.

5.4.1 Birth and death step

The creation and deletion of the marked points is achieved according to the multiple birth and death algorithm [19].

At the birth step a marked point can be created with a predefined probability at every voxel. The parameters of the mark are selected randomly.

After the birth step all of the earlier generated marked points and also the ones which were created in the current iteration are checked. A death rate is calculated from their potential v and they are deleted or left alive according to this value:

$$deathrate = \frac{1}{1 + \kappa e^{-\lambda v}} \tag{5.5}$$

The κ parameter is decreased, λ is increased in each iteration. This step is also stochastic which means that the death rate is compared to a random value and this comparison determines whether the marked point has to be deleted.

5.4.2 Perturbation step

After the birth and death steps there is a perturbation step in every iteration. In this process the parameters of all existing objects are modified randomly. Based on the difference between the object's original potential and the one derived using the newly generated parameters, the modification can be accepted or rejected. If the potential decrease is high enough the probability for accepting the change will also be higher.

5.5 Multilevel operation

As it is mentioned earlier there is a range of possible height and radius values. We have observed that if marked points with any radius in this range can be generated the

smaller ones have a tendency to occupy the place where a larger should be.

This problem can be avoided if not all the possible marked points are created at the same time, but the extraction works on multiple levels. First only larger objects can be generated and the smaller ones are born only on later levels. If the iterations on the current level stop, the marked points are added permanently to the result. On a higher level the lower level objects cannot be changed and cannot be deleted anymore but they are considered when the potentials are calculated. This way the overlap can be avoided and the connections with them are promoted.

The multilevel operation also gives us the possibility that the parameters of the algorithm can be different for each level which can be advantageous.

5.6 Tube refinement

As the postprocessing of the result of the marked point process model I have implemented several methods which can improve the result with the modification of the marked point cylinders. The refinement has three parts: fixing the connections, the direction and the radius of the cylinders.

Although the connections are promoted during the MPP optimization there are usually cylinders which cover the vessels precisely but they are not connected to each other. These gaps can be eliminated with slightly increasing the length of the segments, to check if there is another cylinder close by. The cylinders are increased with maximum 10 voxels in length and the goal is to create an overlap with the connection sphere of another cylinder. If there is an overlap the increase of the length stops. The length will be increased permanently only in that direction where the connection is detected.

The segments which have connections at each end are rotated to maximize the overlap between the end spheres. At most 30° rotation is possible. The orientation is considered the best and will be the new one where the sum of the overlaps of the spheres will be the highest. It is also enforced that existing connections cannot be broken up.

The last step is fixing the inconsistencies between the radii of close by cylinders. The blood vessel networks in the liver are decreased in diameter from one end to the other. This means that the cylinders with connections only at one end should have either small or large radius. Also, the cylinders with connections at both ends should have a radius which is between its neighbors. The doubly connected tubes will have a radius which is equal to the average of the two neighbors. Three voxels large radius was selected as a divider between the small and large cylinders. If a singly connected tube is considered

small but its neighbor is large the radius of the cylinder will be modified to be equal to the connected one's. The modification is similar if the tube is large but the neighboring cylinder is small.

5.6.1 Region growing based on the MPP result

In some cases, for example at curvatures, the cylinders cannot follow the blood vessels correctly and longer ones can have parts which do not cover vessel regions even if the overall fit is good for the MP. This phenomenon implies that smaller units would be better instead of the cylinders, which is possible with using 3D region growing algorithm with seed points based on the tubes.

First the algorithm selects a seed point for each cross section circle of the cylinders. This process is done by finding the maximum intensity voxel in a small region on the plane of the circle. This region is enclosed by the circle which is concentric with the original and its radius is twice as big. A threshold is applied for these voxels, region growing will not be started from low intensity ones. The region growing is performed for every seed point only inside a sphere which has a radius 1.5 times larger than the one of the cross section circle. With executing the algorithm in 3D instead of just in the cross section plane we can avoid leaving out voxels because of rounding errors of the coordinate values.

The intensity threshold for the region growing algorithm is different for tubes in each radius range of the MPP model and was determined with the 5 CT volumes which were also used for parameter tuning.

The maximum intensity voxels used as seed points have more similar directions with the blood vessels than the cylinders had. With fitting a line onto multiple consecutive seed points we can get the new direction vector. This vector is used to find nearby regions at disconnections. If in this direction other positive voxels can be found at a distance of at most 10 voxels, the gap will be filled between them with region growing.

5.7 Results

The 3Dircadb[1] dataset was used for the qualitative and quantitative evaluation of the method. This benchmark consists of 20 contrast enhanced CT volumes from different patients. The majority of them have liver tumours. The slice thickness ranges from 1 mm to 4 mm.

[1] http://www.ircad.fr/research/3dircadb

As it was mentioned earlier the dataset contains ground truth segmentation of the blood vessels done by experts and I have also created a synthetic liver CT image dataset. I have done quantitative evaluation both where five of the volumes were used for tuning the parameters so the remaining 15 pairs were included in the tests and where the parameter tuning was done using 10 synthetic images and for the testing all the images of this database was used obtaining the presented results.

There are some imperfections of the ground truth data from the 3Dircadb dataset, especially several missing vessel segments, which issue was also discussed in [26] and [25]. However, in some cases it is hard to decide where do the vessels end exactly, and the delineation was done by experts regarding the original database. For this reason I decided not to modify the ground truth this way, only the blood vessels outside the liver were removed.

As the dimensions of the images were modified during preprocessing, the results are resized to the original size before evaluation.

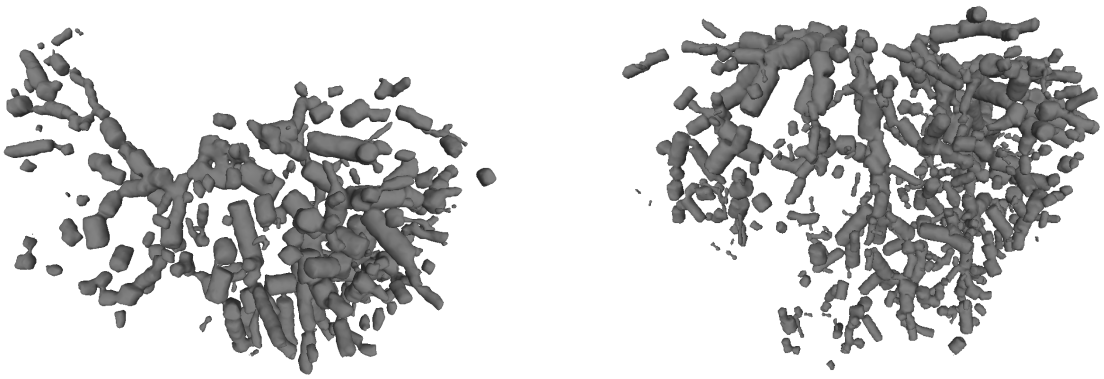

Figure 5.4: 3D mesh result of the tube MPP model without refinement

First I have examined the results of the 3D tube MPP model without refinement and region growing. Examples of these can be seen on Figure 5.4. These initial results are still low precision, as the individual tubes are clearly visible and there are many disconnections between them. It is also visible that many cylinders are very short. This phenomenon is intentional because the gaps and the small incorrectly oriented segments can be corrected effectively in the later steps and these shorter tubes can be fitted onto the blood vessels easier.

Inspecting the refined results slice-by-slice, it can be seen that the edges of the blood vessels are correctly identified, and with the refinement short segments and curves are also accurately delineated. Smaller vessels belonging to the last level of the radius ranges with a diameter of not more than 2 voxels are left out the most often. The region growing

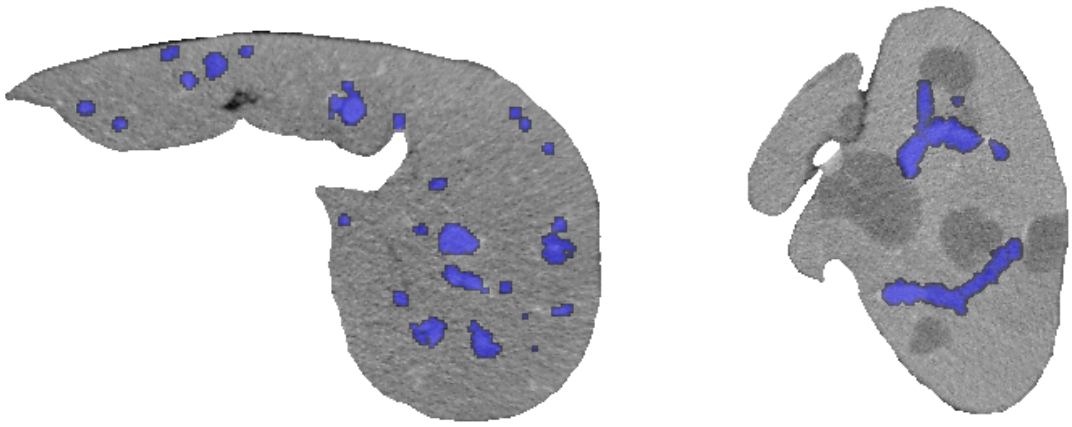

Figure 5.5: Two axial slices of CT volumes with the segmentation result (blue)

algorithm causes the merging of close vessels in rare occasions but the parts of the liver parenchyma are almost never segmented together with the vasculature, only when even the MP cylinder is outside the vessel. The algorithm is robust enough so that tumors inside the liver do not affect the accuracy of the extraction as it is visible on the right side of Figure 5.5.

To visualize the results I have created three dimensional meshes of the extracted blood vessels (Figure 5.6). Comparing these meshes with the tube-based result (Figure 5.4) we can see that the gaps are mostly filled here, and the end result follows the blood vessels much better than the cylinders.

The Sørensen–Dice coefficient, the sensitivity, the specificity and the accuracy were used for quantitative performance evaluation on the 3Dircadb dataset. The results of my method are given in Table 5.1. The first row of results is based on the segmentation where 5 of the 3Dircadb images were used for parameter tuning, the rest 15 for testing. The second row shows testing on these same 15 images the segmenter which was tuned with 10 synthetic images. The last row is similar to this, but here all 20 images of the 3Dircadb dataset were used for testing.

The table shows that although the results of the segmentation are slightly worse in most metrics if the synthetic dataset was used for parameter tuning, this difference is basically nonexistent if we compare the results that are based on the evaluation on the same 15 images. The performance decrease in case of testing on 20 images is likely to be caused by that several of the 5 newly included image have unique histogram which cannot be completely mitigated with the preprocessing.

I have compared the result of segmentation using the real CT image-based tuning

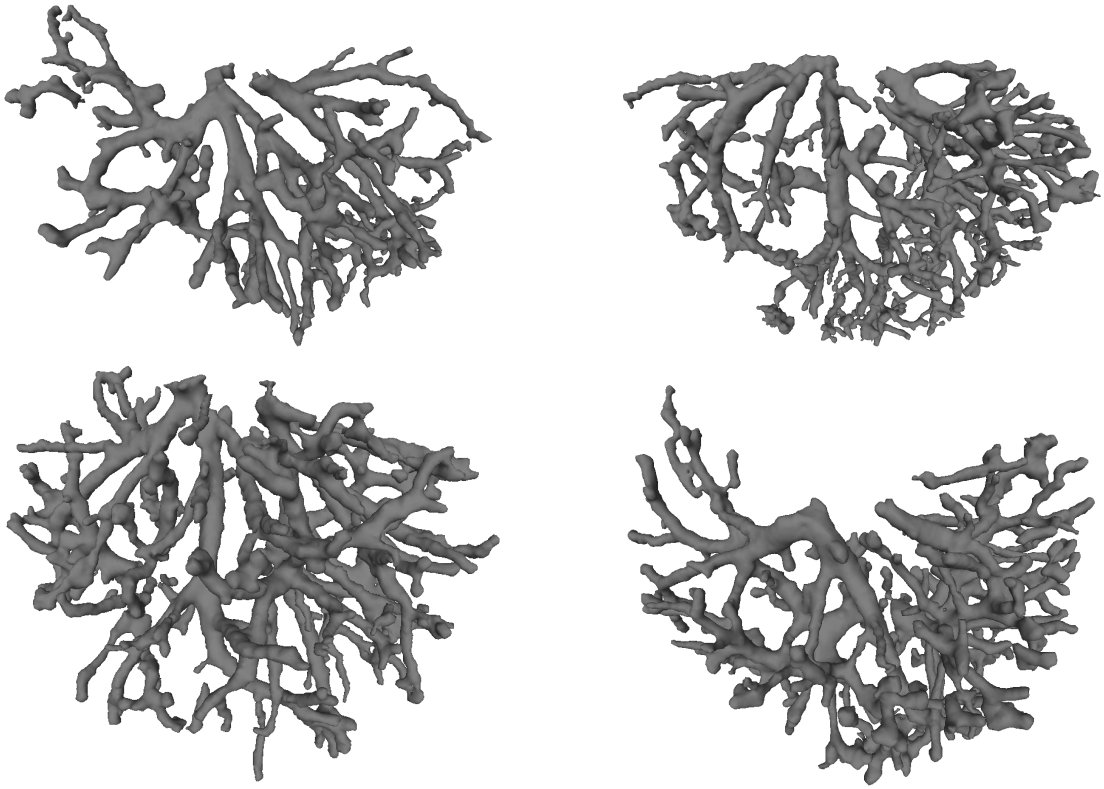

Figure 5.6: 3D meshes of results on the 3Dircadb dataset

with other approaches discussed in Section 3.4.3 where the method was tested also on the 3Dircadb dataset. Although several ones reported slighlty higher Dice rates (0.67-0.69) than my approach, the training and test circumstances were different in each case making direct comparison less relevant. The first technique [24] uses vesselness filter and fuzzy connectedness algorithm, which are considered traditional methods but multiple parameters were set based on the same images which are used for evaluation. The other three considered methods are deep learning techniques. The first one [25] is based on a 3D U-Net architecture. In this work the authors have refined the ground truth (GT) data of the 3Dircadb dataset, but as we have used the original GT, the comparison was done on this non-refined data. [27] utilized a convolutional neural network and performed somewhat better than [24] and [25]. The last method [26], which is also U-Net based, achieved the best quantitative results by far reporting a Dice score around 0.9. However I found multiple concerns regarding the presentation of the results: for example the method proposed in [24] was included here as reference technique with significantly better results than in the original paper. This observation suggests that either the ground truth data was modified or only a small subset of the CT images were used for testing. Also, three quarters of the samples were used for training, and only 25% for the tests,

Tuning	Testing	SDC	Sensitivity	Specificity	Accuracy
5 real	15 real	0.602	0.627	0.976	0.953
10 synthetic	15 real	0.600	0.685	0.967	0.949
10 synthetic	20 real	0.581	0.686	0.967	0.950

Table 5.1: Results on the 3Dircadb dataset

which provides very limited information about the generalization ability of the network. The specificity of my method is higher than the one in [24] and this is also true if we compare the average of the best 10 out of the 15 results with [25] and which measure similar to the result of [27].

As shown in Table 5.1, between the sensitivity of the best and worst results by my approach there is a large difference. In this case only the average of the 5 best results can surpass [24] and [25]. The lower sensitivity values also decrease the SDC and the accuracy but with the improvement of the worst performing CT images the traditional and the first two deep learning method can be surpassed in all four measures.

I have also tested the proposed method on the Visceral[2] and Sliver07[3] datasets but since ground truth data for blood vessels are not included in these benchmarks, quantitative measurements could not be done here. With these datasets we could test how well the algorithm performs with parameters based on CT volumes from the 3Dircadb on other databases. Two example results from these datasets can be seen in Figure 5.7.

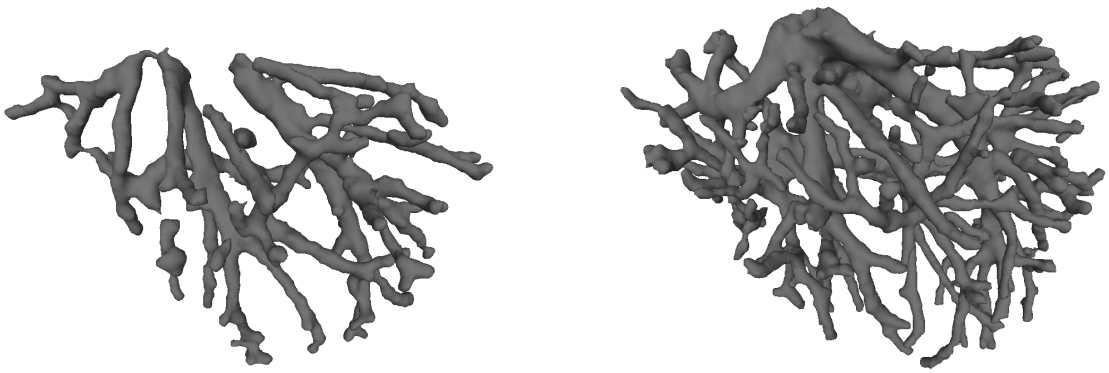

Figure 5.7: 3D mesh result from the Visceral (left) and the Sliver07 (right) dataset

An important characteristic of an algorithm is the required time to perform the segmentation. I have evaluated the speed of the segmentation with measuring the running time in multiple cases. Five different CT image was used as input in these tests and

[2]http://www.visceral.eu/closed-benchmarks/anatomy3/

[3]http://www.sliver07.org/

	Average computation time (s)	Standard deviation of time (s)
MPP first level	134.2	71.00
MPP second level	735.1	383.6
MPP third level	17020	8688
Tube refinement	23.98	20.02
Region growing	14.07	9.896

Table 5.2: Average required time for the steps of the segmentation on 5 CT images

the required time for the main steps of the segmentation were measured separately on an Intel Core i7-3770 CPU. The results are shown in Table 5.2. The first three rows describe the speed of the MPP optimization for different radius ranges. The first level is the fastest as in many cases there is not any vessel segments that are thick enough so that tubes with this large radius are required. The required time is increasing greatly on later levels. This fact can be explained by that the overall length of the thinner vessel segments are larger than the thicker ones', thus more tubes are required to cover these on the image. The MPP-based segmentation time have a high variance on different CT images as the size of the vascular network are different in each liver. The required time for the MPP step can vary on the same CT image because of the stochastic nature of this process. The tube refinement and region growing steps are faster than the MPP optimization. The main factor that determine their speed is the number of the marked points at the end of the last optimization. The whole process takes about 5 hours to complete for one CT image in average.

Several papers of the scientific literature include the evaluation of the required time for the segmentation. In the tests of the active contour model method proposed in [23] the process of locating the whole vascular network of the liver took 130 seconds and 257 seconds in average for the two proposed models. A 3D U-Net neural network-based segmentation was proposed in [25]. The average computation time for the blood vessel segmentation on one computed tomography image was 3.8 minutes. Another deep neural network-based segmentation process was proposed in [26] with computation time evaluation included. The running time was in average 9.45 seconds for this method. These three solutions for hepatic vascular segmentation were faster than my method. It is common in all three cases that GPU calculations were part of the process which fact implies that GPU acceleration is a possible solution for reducing the required time

for the segmentation process. The applicability of these modifications of the algorithm should be tested in the future.

5.8 Testing the segmenter on lungs

The proposed segmentation method is able to adapt to different intensity distribution via the parameter tuning automatically. The adaptation makes it possible to take the differences of the histograms of the livers on CT images into consideration but also makes the method so robust that it is possible that it can perform segmentation on different tree-like structures. For testing this theory I have selected as the goal of the segmentation the vascular network of the lungs.

Some small modifications have to be made for the successful lung vasculature segmentation. One of these is that the program uses 16 bit images in this case, which was required because of the wide intensity range of the lung voxels. If the method would work only with 8 bit images it would result in losing of information. Also, the histogram differences are small in comparison to the contrast inside the lungs. Considering this fact I have skipped the adaptive histogram windowing and used fixed thresholds during preprocessing. In the region growing step larger threshold has to be used in comparison with the liver because of the typical histogram of the lung images and the doubled bit depth but this is determined during tuning together with the MPP parameters.

As public lung vasculature CT dataset was not available for me I have used the synthetic database described in [33]. The lung CT image generator proposed in this paper uses VascuSynth but real lung images are also help in the process to generate realistic lung images. The database contains synthetic CT images and ground truth data for both the bronchial and the vascular network in the liver. For parameter tuning I have used 5 of these images and 15 was included in the testing process.

	SDC	Sensitivity	Specificity	Accuracy
Minimum	0.5574	0.4519	0.9854	0.9731
Maximum	0.6695	0.6718	0.9958	0.9827
Average	**0.6182**	**0.5967**	**0.9895**	**0.9775**

Table 5.3: Results on the 15 lung images of the synthetic dataset

The 15 lung images used for testing were evaluated using the same measures as in case of the liver, these values can be seen in Table 5.3. There is not large difference between

the worst and best performing cases. The Dice coefficient is slightly larger than the one I got for the livers. The specificity is small which means that false positive voxels are rare on the result. Unfortunately, the sensitivity is also small relatively what is caused by that some vessel segments were not found.

Examining the three dimensional models generated from the results, it can be seen that usually the smallest vessels are missing on the results what is expected and are less important to locate than the large vessels.

Figure 5.8: 3D models of two lung vascular network segmentation result (grey) in the lungs (green)

6 Division of the liver into segments

The basis of planning liver resection surgeries is the location of the vascular network and the liver segments which can be determined from the vessels. These segments are functional units, anatomical boundaries cannot be found between them and all of these has their own branch of vein and artery. The most often used models for the segmental division were proposed by Couinaud [2], Goldsmith and Woodburne [3] and Bismuth [4].

The results of the vessel segmentation method that was proposed in the previous chapter can be used for the segmental division of the organ. A branch of the venous network can be selected for each segment and we need to find the corresponding branch points of the network. This task and also finding the end points is achieved with skeletonization. The skeleton image is used to generate a graph representation of the vascular network which idea was proposed in [31]. From the branch points the roots of the correct branches are selected after that and starting from these points and following the skeleton until the end of the tree the whole branch is extracted. The last step is classifying each voxel of the liver volume into one of the segments with finding the closest branch.

6.1 Correcting the root of the venous network

The branches of the veins entering the liver stem from the same vessel which means they should be connected at the root. On the segmentation results it can be seen that this property does not always hold. Upon examining the results I have found that although imperfect segmentation in this area occur but most often the cause of the disconnection is that the root is outside of the liver mask thus ignored in the segmentation process. As the boundaries of the liver mask are not consistently selected the root can be present on some results.

An optional step which can be used if the root is missing from the result is included in my method. The program presents the user a graphical interface which is shown in Figure 6.1. Here the axial slices of the CT scan image and the segmentation result as an overlay can be inspected. The first slider can be used to move between the slices. If

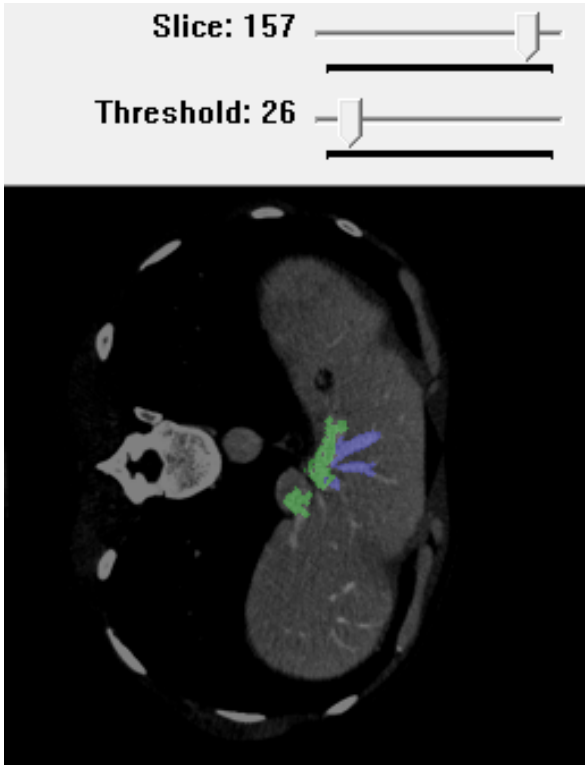

Figure 6.1: The vascular network root corrector interface where the original segmentation result (blue) and the correction (green) is shown on an axial slice of a CT image

the root appear to be missing the user can set a threshold with the second slider and clicking on the slice starts 3D region growing from this point. The algorithm is limited besides by the threshold also by the size of the selected region and it also stops if it reaches the boundary voxels of the segmented vessels. The result of the region growing is visible on the slices and in case if it is incorrect the algorithm can be run again with different seed point and threshold. Using this method after applying hole filling on the region growing segmentation result and combining it with the original result we get the vascular network images together with the roots.

6.2 Skeletonization

Skeletonization is the process of creating a skeleton image by removing voxels from an object on a binary image so that the remaining structure will consist of one voxel wide thin lines. The goal of this method is that the result possess the general shape or topology characteristics of the original object. On the skeleton images it is easier to examine the properties of the objects in these two aspects.

Two main approaches are possible to generate skeleton images. In the first case using distance transformation a distance map is generated first. Here the local extreme points (ridges) are selected to build up the skeleton image. Distance transformation-based skeletonization method was proposed in [34]. In case of complex objects because of the large number of located ridges the skeleton image can be too complex to analyze the structure of the original shape. A method is required to eliminate those ridges that do not carry important information regarding the characteristics of the original object.

To perform the skeletonization I have used the implementation of the 3D thinning algorithm proposed by Lee et al [35] included in the ImageJ[1] software package. This algorithm belongs to the other type of approaches for skeleton image calculation. The idea is that thinning of the object can be performed with removing border points in multiple steps until only a one voxel thin line remains. The so called simple points are border points that do not create or remove holes and connections between the objects, these are the points that are removed. In [35] the method uses an octree representation of a neighborhood of a point for performance improvement with dividing the region into overlapping octants consisting of 2-by-2-by-2 voxels. The octree or adjacency tree that is built up based on the octants contains the center of the neighborhood as the root. Each non-terminal node has eight children and leaves are the neighboring points. Six directions are considered: north, south, east, west, up and down and in a single step only the neighbors in that direction are checked to determine whether the current voxel is a border point or not. After each directions are checked and the located simple points are removed the process is restarted and this iteration is repeated until more points can not be removed.

6.3 Finding end and branch points

The goal of the skeleton image analysis is finding end and branch points of the vascular network. This task can be done with inspecting the local neighborhood of the voxels.

Multiple neighborhood types can be considered in two and three dimension when talking about connections. In 2D usually 4-connectivity or 8-connectivity is used. The first includes only the horizontal and vertical neighbors, the second does diagonal ones also. In 3D the simplest is the 6-connectivity where only the voxels with a common face with the examined voxel considered. 18-connectivity also includes neighbors where only an edge is common, in case of 26-connectivity besides the faces and edges, the corners

[1] https://imagej.net/

can also be common for the voxel to be considered. The voxels of the vascular network skeleton image can be connected in any direction that is why I have used 26-connectivity.

The fact that the skeleton of an object contains one voxel wide lines means that in term of the number of the neighbors four cases are possible. If a voxel has no neighbors it is a detached voxel and can be removed. In case of one neighbor the voxel is considered to be an end point. If the voxels has exactly two neighbors it is an intermediary point on a line. The last case is when there are more than two neighbors. This property means that more than two paths are possible to leave the current voxel which implies that it is a branch point.

6.4 Graph building

After the end and branch points are selected their connections with each other are determined. When all the connections are discovered a graph is built from this information which represents the vascular network.

The first step is finding the branch point which the end point is connected to. Here only one path has to be traced, all the intermediary voxels has two neighbors. Starting from an end point the neighbors of the current voxel are collected and the algorithm steps onto the voxel that is not on the mapped part of the path yet. There is a special case where this way the method reaches another end point, these are segments are disconnected from the network thus can be removed.

The next step is mapping the paths from each branch point. A branch point has at least three neighbors but it can be assumed that these neighboring voxels has at most two. These properties imply that the branch point neighbors has to be collected first but after that the path tracing algorithm is basically the same as the one described in the previous paragraph, it walks through the neighbors to an end or branch point. The paths between the end points and the first branch point are traversed twice but these duplicates will be removed later.

At this state only direct connections between the end points and the branch points are known, there is no information about the relationship between distant points. Starting from an end point using breadth-first traversal the tree is mapped, the nodes, edges and the exact paths belonging to them are stored. Every end point is checked but traversal started only from the ones which was not visited yet, this way duplication of the trees is avoided. The result of this step will be the vascular network represented in form of a forest consisting of one or multiple trees.

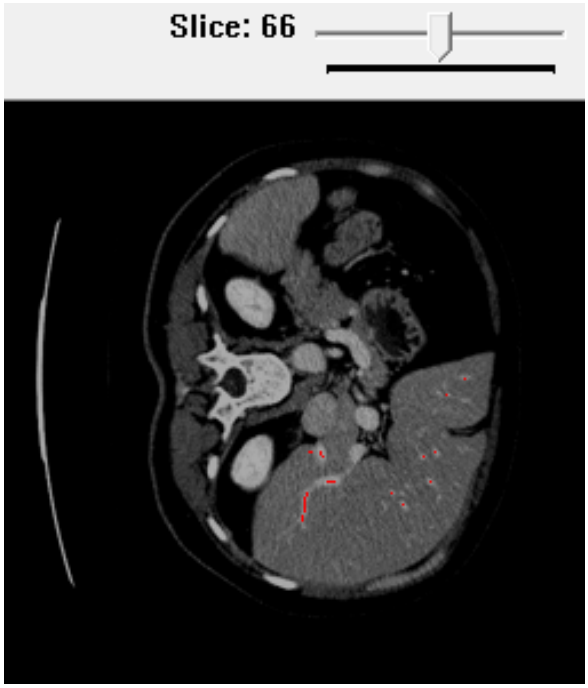

Figure 6.2: The approximate root point selector interface where the skeleton image (red) is shown on an axial slice of a CT image

6.5 Selecting a root point

The layout of the vascular network is determined but its root point at this stage is unknown. One possibility is to use a root point that was already selected by the user earlier but as this step was optional it is better to use another interface for this task only.

The graphical interface (Figure 6.2) is very similar to the previous one. The threshold slider is not necessary here thus it is removed and instead of the segmentation result the skeleton image is overlaid onto the CT scan image. With the help of the skeleton image the user can select the approximate root point of the venous network. The algorithm locates the closest branch point of the skeleton image which will be the root point.

6.6 Extracting subtrees

In this step the branch points of the venous network which are the root of the subtrees of the liver segments are selected. Locating these branch points is done by starting from the root point of the whole network, the subtrees that start at the connected nodes are inspected based on their size. It is important to note that the size is determined based on the number of skeleton points and not the number of the nodes, the former calculation is more representative about the real size of the segment.

There are three categories of subtrees based on their size. If the size of the subtree is very small compared to the size of the whole tree the subtree is removed. On the contrary, if the size ratio is too large the subtree is divided onto smaller subtrees with moving to the connected nodes and setting them as root for the new subtrees. In this process the algorithm never selects a connected node that was already checked because this would mean stepping back and checking an even larger already dismissed subtree. If the size ratio of a subtree and the whole tree is between these two categories the subtree is accepted and selected as a segment defining subtree. The exact number of segments is not known but at least 8 is expected and can be more [5]. Based on this fact the threshold of the size ratio of a too large subtree was set to 15%. A subtree is considered too small if its size ratio is under 3%.

The voxels of the exact paths of the subtrees are saved and used in the next step.

6.7 Assigning liver voxels to segments

In the previous steps all the skeleton points of each subtree where collected. The liver voxels are assigned to a segment. The segmentation of the liver itself was not part of this thesis and I have used the liver annotation data included in the testing datasets to determine the location of the liver voxels. This task is achieved by iterating through the liver voxels and finding the closest skeleton point to them.

6.8 Results

I have evaluated the method on 10 CT images of the 3Dircadb dataset. The segmentation results of the algorithm proposed in Chapter 5 were used as input. First I have corrected the vascular network root regions using the graphical interface and after the skeleton image has been calculated I have set the approximate location of the root point of the network. The resulting images have the same size as the CT images and liver voxels belonging to the same segment have the same intensity.

Locating the segments was successful on 8 of the 10 images. In one case artifacts on the CT image caused that on the vessel segmentation result in one part of the liver the ratio of false positive voxels was higher in comparison with the remaining region. This phenomenon caused that in the affected area more, in the other parts less subtrees were located than expected. The other unsuccessful segment delineation has happened because the generated skeleton image was incorrect. In some regions instead of the

centerline voxels close to the original boundary of the vessels were selected as skeleton points. This result caused that the algorithm was unable to find the correct path and determine the the actual size of the subtrees. These two cases indicate that the robustness of the process should be improved later.

In the 8 successful cases a varying number of segments with similar volumes were located. At least 8 segments were found and the maximum was 11. Examination of the results shows that when a subtree considered too small was removed these segments were caused by imperfectly segmented vessel boundaries or very small vessel segments. These observations indicate that removing these subtrees were justified. In some cases I have observed that one subgraph was too small and this fact has caused that a neighboring segment occupied a region of the segment of this smaller subgraph.

	Average computation time (ms)	Standard deviation of time (ms)
Correcting root region	19.28	2.179
Finding end and branch points	1240	651.6
Graph generation	6.429	2.219
Extracting subgraphs	163.7	151.0
Liver voxel classification	5710	3997

Table 6.1: Average required time for the steps of the algorithm on 5 CT images

Similarly to segmentation I have evaluated the speed of the liver segment locator on the same CPU. The average required time on 5 CT images for each step of the process are shown in Table 6.1. The speed of the skeletonization of the vascular network image is not included as this process is preformed by an external software. The slowest steps are the liver voxel classification and locating the principal points on the skeleton. The overall time required for locating the segments depends on the user but in my test it was under a minute in each case.

The results were not evaluated quantitatively because CT image datasets with high quality liver segment annotation are not publicly available. In the scientific literature it was shown that evaluation can be done on liver corrosion casts where the liver tissue is removed chemically and only the vascular network remains and these casts are scanned using computer tomograph. In case of the availability of such CT images quantitative evaluation of the results can be done in the future.

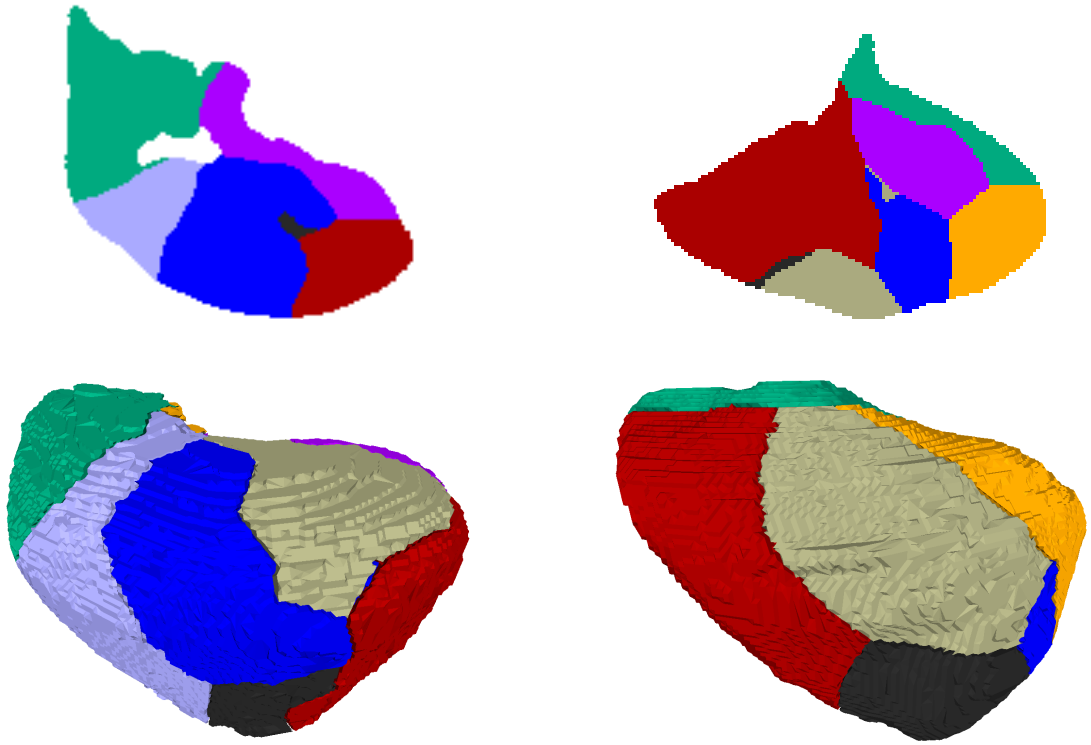

Figure 6.3: Liver segment delineation results on axial slices (first row) and on 3D models of the liver (second row); each color represent a different segment

7 Conclusion

The treatment of multiple liver related illnesses include the surgical resection of the organ. In case of liver tumor or transplantation a part of it is removed and if the surgery was successful the remaining portion can fulfill its role. Because of the functions of the liver its whole volume is intervened by blood vessels so that the blood supply is ensured for the cells and also the detoxification can be carried out. During the resection the most important task is based on the location of the vessels determining the placement of the cut so that after the surgery the remaining part will stay functional.

Contrast enhanced computed tomography is a suitable imaging technique to locate blood vessels because both the resolution and the contrast is sufficient to classify the vessel and non-vessel voxels. The 3D imaging makes three dimensional segmentation possible, the whole organ can be processed simultaneously. Marked point process model based image segmentation methods used extensively in 2D to find unknown number of objects with similar characteristics but their 3D application is rare. One of their drawbacks is that the successful segmentation depends on the fine tuning of multiple parameters which is done manually. Another disadvantage of these methods is that the general shape of the marked points cannot follow perfectly follow the boundaries of real world objects. Region growing is a simple segmentation algorithm which was even proposed for the use of liver vessel segmentation but achieving fully automatical operation with good results is a hard task and also, due to the differences of the thicker and thinner vessel segments on CT images, false positive voxels are frequent.

In this work I have developed a segmentation method which combines an MPP model based and region growing segmentation steps where the MPP model works in 3D, its parameters are selected automatically without the need of training data consisting of real CT images and manual annotation. The result of this step is improved with three dimensional region growing performed on small regions started from automatically selected seed points and the threshold is also set without the need of human interaction. With the combination of these algorithms the drawbacks of them can be greatly or even entirely mitigated and I could get a much better result this way.

The results were evaluated on three CT image datasets, quantitatively on scans of the 3Dircadb database comparing them to the manually segmented blood vessels.

The last step of the development process was finding the segments of the liver. In this thesis I have proposed a method that can perform this task on annotated vascular network images created by my segmentation algorithm selecting the root of venous network by hand first, then with the help of skeletonization creating a graph representation of its tree. After selecting the root branch points of the subtrees each of that belong to a segment the algorithm extracts the subtree itself. Based on the points of the subtrees each voxel of the liver are classified to one of the segments.

The evaluation of the methods shows that further improvement and clinical evaluation is required but their application for liver resection surgery planning will be possible.

8 Future work

The proposed vessel segmentation results were evaluated extensively and were tested on livers with anatomical modifications. The algorithm can handle liver tumors but in case of high intensity artifacts on the CT image it performs worse. I am planning to solve this problem in the future.

The speed evaluation of the segmentation algorithm has revealed that the required time for the process has to be decreased. Solutions for this problem include precalculating some of the three dimensional geometrical objects that are created in each iteration of the optimization and accelerating the algorithm with preforming calculations on a GPU. The speed increase using these possible solutions has to be evaluated and in case these modifications fulfill their roles they has to be implemented later.

The skeleton generation from the vessel network image was inaccurate in one case. I will inspect the input image and the generator algorithm also and compare it to other skeletonization methods. The best performing algorithm has to be implemented and included in the program so that external software will not be required.

I am also planning to evaluate quantitatively the segment delineation algorithm and compare it to the results of other methods. Based on the result I will improve the proposed method in those aspects where it performs worse.

9 Acknowledgements

I would like to thank my supervisor, Dr. Csaba Benedek the provided feedback and guidance throughout my work and the Institution for Computer Science and Control for providing the tools required for this project.

This thesis work was supported by the National Research, Development and Innovation (NRDI) Office within the frameworks of the Autonomous Systems National Laboratory, and the Artificial Intelligence National Laboratory programs, and by the NRDI Fund (OTKA) grant number K-120233.

References

[1] J. Szentágothai and M. Réthelyi, *Funkcionális anatómia II.* Medicina, 2006.

[2] C. Couinaud, *Le foie: études anatomiques et chirurgicales*. Masson, 1957.

[3] M. Goldsmith, "The surgical anatomy pertaining to liver resection," *Surg Gynecol Obstet*, vol. 105, pp. 310–316, 1957.

[4] H. Bismuth, "Surgical anatomy and anatomical surgery of the liver," *World journal of surgery*, vol. 6, no. 1, pp. 3–9, 1982.

[5] J. H. Fasel, P. E. Majno, and H.-O. Peitgen, "Liver segments: An anatomical rationale for explaining inconsistencies with couinaud's eight-segment concept," *Surgical and radiologic anatomy*, vol. 32, no. 8, pp. 761–765, 2010.

[6] P. J. Green, "Reversible jump markov chain monte carlo computation and bayesian model determination," *Biometrika*, vol. 82, no. 4, pp. 711–732, 1995.

[7] G. P. Otto and T. K. Chau, "Region-growing algorithm for matching of terrain images," *Image and vision computing*, vol. 7, no. 2, pp. 83–94, 1989.

[8] S. Park, J. Lee, H. Lee, *et al.*, "Parallelized seeded region growing using cuda," *Computational and mathematical methods in medicine*, vol. 2014, 2014.

[9] M. Danu, C.-I. Nita, A. Vizitiu, C. Suciu, and L. M. Itu, "Deep learning based generation of synthetic blood vessel surfaces," *2019 23rd International Conference on System Theory, Control and Computing (ICSTCC)*, 2019. DOI: 10.1109/icstcc.2019.8885576.

[10] M. A. Galarreta-Valverde, M. M. G. Macedo, C. Mekkaoui, and M. P. Jackowski, "Three-dimensional synthetic blood vessel generation using stochastic l-systems," *Medical Imaging 2013: Image Processing*, 2013. DOI: 10.1117/12.2007532.

[11] G. Hamarneh and P. Jassi, "Vascusynth: Simulating vascular trees for generating volumetric image data with ground-truth segmentation and tree analysis," *Computerized Medical Imaging and Graphics*, vol. 34, no. 8, pp. 605–616, 2010. DOI: 10.1016/j.compmedimag.2010.06.002.

[12] F. Lafarge, G. Gimelfarb, and X. Descombes, "Geometric feature extraction by a multimarked point process," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1597–1609, 2010. DOI: 10.1109/tpami.2009.152.

[13] T. Li, M. Comer, and J. Zerubia, "Feature extraction and tracking of cnn segmentations for improved road detection from satellite imagery," *2019 IEEE International Conference on Image Processing (ICIP)*, 2019. DOI: 10.1109/icip.2019.8803355.

[14] B. Keresztes, O. Lavialle, S. Pop, and M. Borda, "Fiber segmentation on composite materials using marked point processes," *Acta Technica Napocensis*, vol. 50, 2009.

[15] T. Li, M. Comer, and J. Zerubia, "An unsupervised retinal vessel extraction and segmentation method based on a tube marked point process model," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020. DOI: 10.1109/icassp40776.2020.9054023.

[16] ——, "A connected-tube mpp model for object detection with application to materials and remotely-sensed images," *2018 25th IEEE International Conference on Image Processing (ICIP)*, 2018. DOI: 10.1109/icip.2018.8451108.

[17] W. Ge and R. T. Collins, "Marked point processes for crowd counting," *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009. DOI: 10.1109/cvpr.2009.5206621.

[18] Y. Yu, J. Li, H. Guan, C. Wang, and M. Cheng, "A marked point process for automated tree detection from mobile laser scanning point cloud data," *2012 International Conference on Computer Vision in Remote Sensing*, 2012. DOI: 10.1109/cvrs.2012.6421249.

[19] Z. Németh and C. Benedek, "Automatic tumuli detection in lidar based digital elevation maps," *XXIV ISPRS Congress*, 2020.

[20] S. Roychowdhury, D. Koozekanani, and K. Parhi, "Blood vessel segmentation of fundus images by major vessel extraction and sub-image classification," *IEEE Journal of Biomedical and Health Informatics*, pp. 1–1, 2014. DOI: 10.1109/jbhi.2014.2335617.

[21] Y. Yang, S. Huang, and N. Rao, "An automatic hybrid method for retinal blood vessel extraction," *International Journal of Applied Mathematics and Computer Science*, vol. 18, no. 3, pp. 399–407, 2008. DOI: 10.2478/v10006-008-0036-5.

[22] S. Badsha, A. W. Reza, K. G. Tan, and K. Dimyati, "A new blood vessel extraction technique using edge enhancement and object classification," *Journal of Digital Imaging*, vol. 26, no. 6, pp. 1107–1115, 2013. DOI: 10.1007/s10278-013-9585-8.

[23] Y.-Z. Zeng, S.-H. Liao, P. Tang, *et al.*, "Automatic liver vessel segmentation using 3d region growing and hybrid active contour model," *Computers in Biology and Medicine*, vol. 97, pp. 63–73, 2018. DOI: 10.1016/j.compbiomed.2018.04.014.

[24] R. Zhang, Z. Zhou, W. Wu, C.-C. Lin, P.-H. Tsui, and S. Wu, "An improved fuzzy connectedness method for automatic three-dimensional liver vessel segmentation in ct images," *Journal of Healthcare Engineering*, vol. 2018, pp. 1–18, 2018. DOI: 10.1155/2018/2376317.

[25] Q. Huang, J. Sun, H. Ding, X. Wang, and G. Wang, "Robust liver vessel extraction using 3d u-net with variant dice loss function," *Computers in Biology and Medicine*, vol. 101, pp. 153–162, 2018. DOI: 10.1016/j.compbiomed.2018.08.018.

[26] Q. Yan, B. Wang, W. Zhang, *et al.*, "An attention-guided deep neural network with multi-scale feature fusion for liver vessel segmentation," *IEEE Journal of Biomedical and Health Informatics*, pp. 1–1, 2020. DOI: 10.1109/jbhi.2020.3042069.

[27] M. Xu, Y. Wang, Y. Chi, and X. Hua, "Training liver vessel segmentation deep neural networks on noisy labels from contrast ct imaging," *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, 2020. DOI: 10.1109/isbi45749.2020.9098509.

[28] H.-P. Meinzer, M. Thorn, and C. E. Cárdenas, "Computerized planning of liver surgery—an overview," *Computers & Graphics*, vol. 26, no. 4, pp. 569–576, 2002.

[29] C. Zahlten, H. Jürgens, and H. Peithen, "Reconstruction of branching blood vessels from ct-data," *Visualization in scientific computing*, pp. 41–52, 1995.

[30] X. Yang, J. Do Yang, H. P. Hwang, *et al.*, "Segmentation of liver and vessels from ct images and classification of liver segments for preoperative liver surgical planning in living donor liver transplantation," *Computer methods and programs in biomedicine*, vol. 158, pp. 41–52, 2018.

[31] D. Selle, B. Preim, A. Schenk, and H.-O. Peitgen, "Analysis of vasculature for liver surgical planning," *IEEE transactions on medical imaging*, vol. 21, no. 11, pp. 1344–1357, 2002.

[32] A.-C. Feng, H.-L. Fan, T.-W. Chen, and C.-B. Hsieh, "Hepatic hemodynamic changes during liver transplantation: A review," *World Journal of Gastroenterology*, vol. 20, no. 32, p. 11 131, 2014. DOI: 10.3748/wjg.v20.i32.11131.

[33] D. Jimenez-Carretero, R. S. J. Estepar, M. D. Cacio, and M. J. Ledesma-Carbayo, "Automatic synthesis of anthropomorphic pulmonary ct phantoms," *Plos One*, vol. 11, no. 1, 2016. DOI: 10.1371/journal.pone.0146060.

[34] F. Y. Shih and C. C. Pu, "A skeletonization algorithm by maxima tracking on euclidean distance transform," *Pattern Recognition*, vol. 28, no. 3, pp. 331–341, 1995.

[35] T.-C. Lee, R. L. Kashyap, and C.-N. Chu, "Building skeleton models via 3-d medial surface axis thinning algorithms," *CVGIP: Graphical Models and Image Processing*, vol. 56, no. 6, pp. 462–478, 1994.