Ph.D. Dissertation

Liver and Vessel Segmentation on Abdominal CT Images

복부 CT에서 간과 혈관 분할 기법

February 2020

Department of Computer Science and Engineering
The Graduate School
Seoul National University

Minyoung Chung

Liver and Vessel Segmentation on Abdominal CT Images

지도교수 신 영 길

이 논문을 공학박사 학위논문으로 제출함

2019 년 11 월

서울대학교 대학원

컴퓨터공학부

정 민 영

정민영의 공학박사 학위논문을 인준함

2019 년 12 월

위 원 장 ______김 명 수______ (인)

부위원장 ______신 영 길______ (인)

위　　원 ______김 건 희______ (인)

위　　원 ______김 보 형______ (인)

위　　원 ______이 정 진______ (인)

Abstract

Accurate liver and its vessel segmentation on abdominal computed tomography (CT) images is one of the most important prerequisites for computer-aided diagnosis (CAD) systems such as volumetric measurement, treatment planning, and further augmented reality-based surgical guide. In recent years, the application of deep learning in the form of convolutional neural network (CNN) has improved the performance of medical image segmentation, but it is difficult to provide high generalization performance for the actual clinical practice. Furthermore, although the contour features are an important factor in the image segmentation problem, they are hard to be employed on CNN due to many unclear boundaries on the image. In case of a liver vessel segmentation, a deep learning approach is impractical because it is difficult to obtain training data from complex vessel images. Furthermore, thin vessels are hard to be identified in the original image due to weak intensity contrasts and noise. In this dissertation, a CNN with high generalization performance and a contour learning scheme is first proposed for liver segmentation. Secondly, a liver vessel segmentation algorithm is presented that accurately segments even thin vessels.

To build a CNN with high generalization performance, the auto-context algorithm is employed. The auto-context algorithm goes through two pipelines: the first predicts the overall area of a liver and the second predicts the final liver using the first prediction as a prior. This process improves generalization performance because the network internally estimates shape-prior. In addition to the auto-context, a contour learning method is proposed that uses only sparse contours rather than the entire contour. Sparse contours are obtained and trained by using only the mispredicted part of the network's final prediction. Experimental studies show that the proposed network is superior in accuracy to other modern networks. Multiple N-fold tests are also performed to verify the generalization performance.

An algorithm for accurate liver vessel segmentation is also proposed by introducing vessel candidate points. To obtain confident vessel candidates, the 3D image is first reduced to 2D through maximum intensity projection. Subsequently, vessel segmentation is performed from the 2D images and the segmented pixels are back-projected into the original 3D space. Finally, a new level set function is proposed that utilizes both the original image and vessel candidate points. The proposed algorithm can segment thin vessels with high accuracy by mainly using vessel candidate points. The reliability of the points can be higher through robust segmentation in the projected 2D images where complex structures are simplified and thin vessels are more visible. Experimental results show that the proposed algorithm is superior to other active contour models.

The proposed algorithms present a new method of segmenting the liver and its vessels. The auto-context algorithm shows that a human-designed curriculum (i.e., shape-prior learning) can improve generalization performance. The proposed contour learning technique can increase the accuracy of a CNN for image segmentation by focusing on its failures, represented by sparse contours. The vessel segmentation shows that minor vessel branches can be successfully segmented through vessel candidate points obtained by reducing the image dimension. The algorithms presented in this dissertation can be employed for later analysis of liver anatomy that requires accurate segmentation techniques.

Keywords: Active contour model, auto-context neural network, contour attention, liver segmentation, vessel candidates, vessel segmentation.

Student Number: 2014-21778

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and motivation

Medical image analysis is consistently gaining its demand because the number of medical images has been growing since the X-ray imaging technology had been developed. A computed tomography (CT) reconstruction from X-ray images opened a new era of accurate computer-aided diagnosis (CAD) based on 3-dimensional (3D) images. Medical image segmentation is one of the most important and essential prerequisites for clinical applications (i.e., radiological diagnosis system) of the automated CAD such as disease diagnosis, treatment planning, volume measurement, and further virtual/augmented surgeries [1, 2] (Fig. 1.1). Among the organs, the liver is a highly demanded organ where its disease is one of the top increasing causes of death worldwide. For accurate surgical planning such as liver transplantation and resection, volumetric information and vascular structure analysis of a liver is critically required. However, due to its time-consuming labor of manual 3D image annotation, computer-

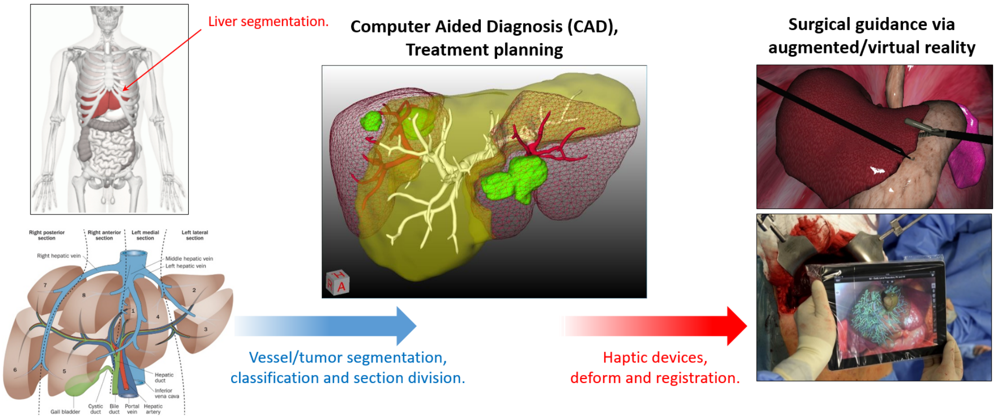


Figure 1.1: Importance of liver segmentation. Liver segmentation is a critical prerequisite for many further applications such as vessel/tumor segmentation, liver sectioning, treatment planning, and surgical simulations. The sources of figures are noted in footnotes[1].

aided surgical planning is highly limited.

In recent years, empowered by cutting edge hardware infrastructures, large-scaled medical image data is available to employ an artificial intelligence towards CAD system. However, most of the artificial intelligence is constructed in a supervised manner which implies the importance of manual ground-truth image annotations. As aforementioned, manual annotation of 3D medical images is a tedious task. The degree of difficulty increases even more for very complex objects, such as liver vessels. Therefore, it is necessary to develop artificial intelligence that shows high generalization performance through a small number of annotated data or to introduce an accurate segmentation method by human-designed algorithms.

[1] https://en.wikipedia.org/wiki/Liver

[1] https://www.nibib.nih.gov/news-events/newsroom

[1] https://9to5mac.com/2013/08/21/liver-surgery-now-theres-an-ipad-app-for-that

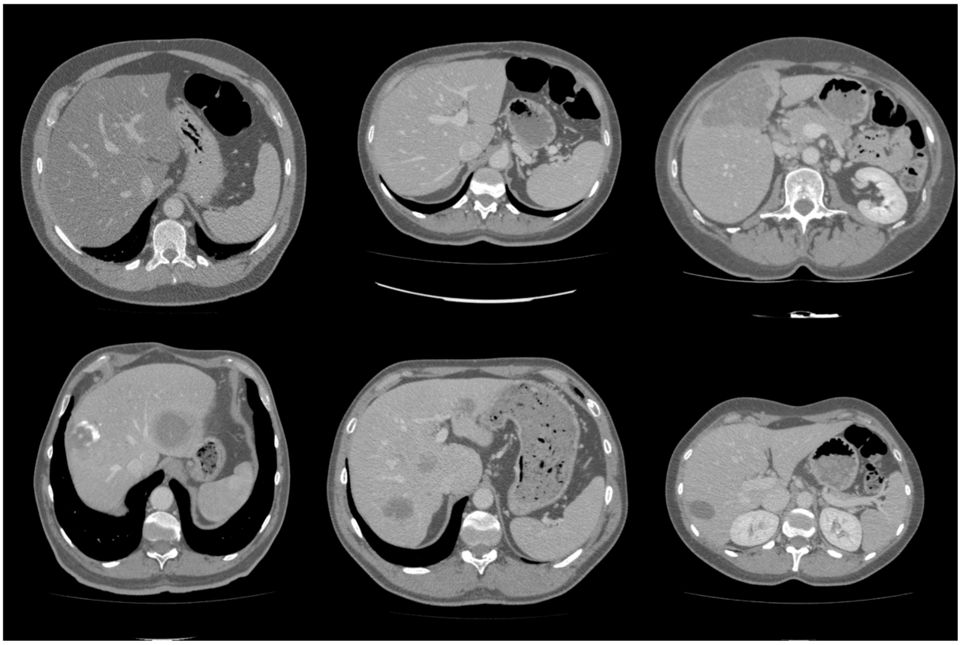


Figure 1.2: Dynamic intensity variations in liver regions. Abdominal CT images contain an intrinsic class imbalance in the intensity distribution due to multi-phases or rare cases of anomalies such as tumors.

1.2 Problem statement

Manual or semi-automatic segmentation of a liver and vessels is a very impractical task owing to its large shape variability, unclear boundaries, and complex structure. Unlike other organs, ambiguous boundaries with the heart, stomach, pancreas, and fat make liver segmentation difficult. Furthermore, manual segmentation is error-prone which implies there is a severe inter- and intra-observer variability of the results.

Although the CNN-based methods are showing groundbreaking results compared to the classics, the performance of generalization should be addressed for the actual employment of CNNs for medical image segmentation task. As shown

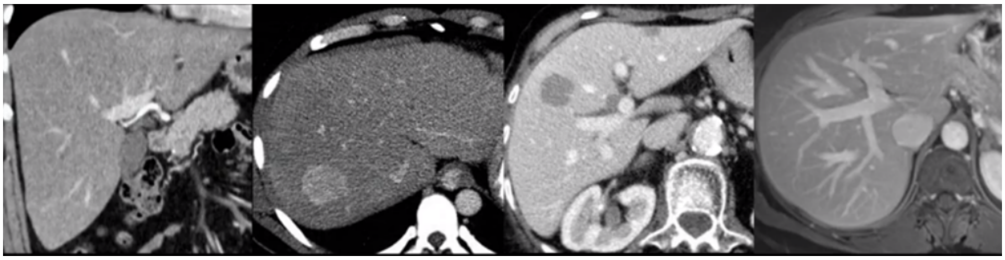
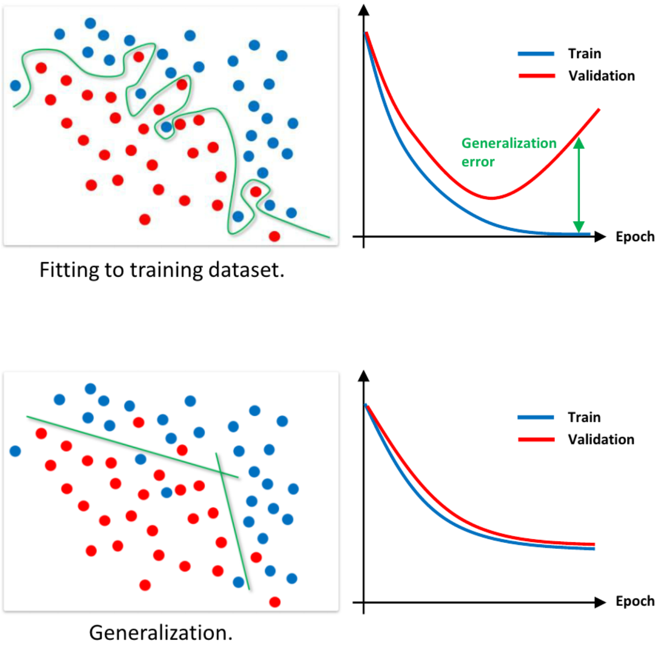


Figure 1.3: Multiple phases of abdominal CT imaging.

in Fig. 1.4, a neural network has a strong capability (i.e., complexity) to classify arbitrary signals mainly owing to deep and complex stacks of trainable parameters. Ironically, the main difficulty of deep neural networks to be employed in the clinics arises from its strong capability. As a huge number of parameters are trainable, deep neural networks opt to fit the training dataset. In other words, networks are easily over-fitted to the training dataset. It is very hard to make a network to be trained for general cases unless a large number of training images are provided. Many studies have been conducted to obtain a high generalization performance such as weight decay, drop out [3], transfer learning [4], data augmentation [5], domain adaptation [6, 7], and regularization of loss functions [8]. However, those systematic techniques have limitations to be adapted in various fields that have data deficiency and intrinsic class imbalance (e.g., rare cases of anomalies and phases in medical images (Figs. 1.2 and 1.3)). Consequently, a domain-specific generalization technique is highly required especially in the field of medical image analysis.

In the image segmentation problem, it is worth knowing that an object boundary delineation is the most effective and accurate way for object segmentation. There has been a huge body of literature to resolve accurate contour delineation for object segmentation [9]. An implantation of contour features to a neural network has been previously studied in [10]. However, the difficulty of



Fitting to training dataset.

Generalization.

Figure 1.4: Capability of deep neural networks (upper left) and the corresponding generalization error (upper right). Train and validation errors while training the neural networks are schematically plotted. The lower left classifier shows the best performance of generalization.

contour delineation of a liver, unlike other organs, is that boundary features of the ground-truth liver are typically irregular (Fig. 1.5). It is very difficult to explicitly model the features of whole boundaries as opposed to the reference [10]. That is, a complete delineation of a contour is hard to be trained even for the neural network.

Segmentation of liver vessels is even more difficult than that of a liver from the perspective of its complex structure and limited annotations. Blood vessels typically have tree-structure with continuous tubular sections. Tubular and branching tree structures of vessels are an anatomical nature of the human

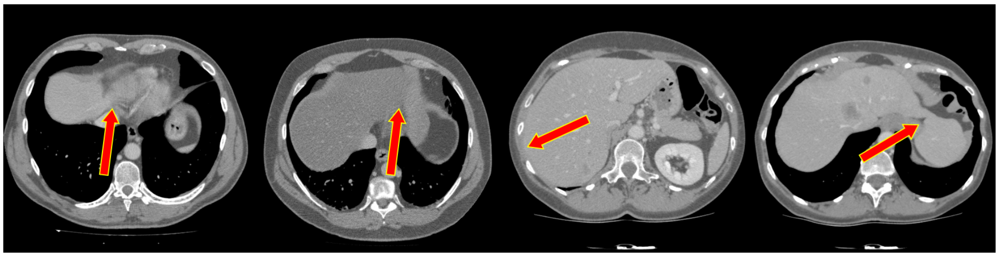


Figure 1.5: Ambiguous boundaries on a liver. It is very hard to identify its boundaries based on a local intensity analysis.

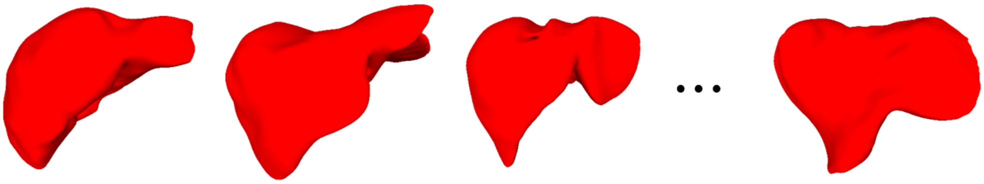


Figure 1.6: Diverse shapes of liver.

vascular system. However, algorithms assuming structural properties of vessel might break down in some patients. For example, vascular structures are not fully contrast-enhanced in the same phase in every patient, and vessel structures may be broken by vascular disease or cancerous regions. These individually distinctive situations make it difficult to segment vessels automatically. Moreover, thin vessels are hard to be identified in the original 3D image which makes it difficult to accurately segment all vessels.

1.3 Main contributions

In this dissertation, a neural network-based architecture is first proposed that performs an accurate segmentation of a liver on abdominal CT images. A human-designed curriculum is employed while training the network. The network initially estimates an overall shape of a liver, and subsequently delineates fine details. The proposed two-stage prediction showed high generalization performance without any extra techniques. Besides, a contour scheme is successfully embedded in the network to improve accuracy. Secondly, a fully automated algorithm for liver vessel segmentation is proposed. The prior of a liver (i.e., liver segmentation), which is obtained by the proposed neural network, is employed to the vessel segmentation algorithm. Minor vessels, that are hard to be identified in the original CT volume, are successfully segmented by introducing a novel 3D map of vessel candidates. A brief overview of algorithms and achievements are described in the following paragraphs.

The overall shape estimation and localization is the most challenging and important task for the generalized performance of liver segmentation because the variability of shape is extremely severe (Fig. 1.6). The proposed liver segmentation method employs an auto-context algorithm [11] into the neural network. The auto-context algorithm [11] is formulated by a single neural network by using a liver prior branch. The liver prior branch is deeply supervised to generate the probability of a liver foreground. Effective high-level residual connections were applied for the liver-prior estimation. The prior is then fused with deep contexts for the final auto-context layers. In addition to the auto-context structure, another branch was added which is also deeply supervised by contours of a liver. Instead of training the explicit ground-truth contour, more significant sparse contours are trained which act as an implicit attention that can improve the final delineation of the target liver object. The sparse contours are obtained and trained based on self-supervising fashion by using only the mispredicted part of the network's final prediction. The objective of learning partially significant contours is that, unlike other segmentation problems (e.g., glands), the contour of a liver is difficult to be obtained accurately, even with deep CNNs, because of its ambiguous boundaries (Fig. 1.5). The main underlying principle of the proposed architecture is that accurate segmentation of a liver can be achieved by a robust shape-prior and an accurate delineation of a contour region. The aforementioned challenges were accomplished by the two architectures: the high-level residual shape-prior estimation in an auto-context framework and the self-supervised contour attention. The robust shape-prior enhanced the performance of generalization, and the contour attention mechanism improved the final accuracy. Experimental results, comparing with many state-of-the-art neural networks, show the supremacy of the proposed method. Several ablation studies have been conducted to verify each designed concept (i.e., auto-context and contour). Additionally, the performance of generalization is assessed in-depth based on multiple N-fold validations.

For the task of vascular structure analysis, a fully automated liver vessel segmentation algorithm is proposed including portal and hepatic veins on contrast-enhanced CT images. First, vessel candidate points are extracted from a 3-dimensional (3D) CT image. To generate accurate points, the 3D segmentation problem is reduced to a 2D problem by generating multiple maximum intensity images based on slabbed regions (i.e., depth-constrained regions). After performing vessel segmentation on maximum intensity images, the foreground pixels are back-projected to the original 3D space. A large set of maximum intensity images produces a very dense and accurate vessel candidate map. Finally, a newly designed active contour model is proposed for an accurate segmentation of vessels. The model encompasses the original image, vessel probability map from dense vessel candidates, and a good prior of an initial contour. In total, 55 abdominal CT images are used for a parameter study and a quantitative evaluation. The performance of the proposed method is evaluated by comparing it with other state-of-the-art active contour models for vascular images applied directly to the original image. The result showed that the proposed method successfully segmented vascular structures 25%-122% more accurately than other methods without any extra false positive segmentation. The proposed model can generate a smooth and accurate boundary of a vessel object and easily extract thin and weak peripheral branch vessels. The detailed result can aid further anatomical studies such as structural analysis of hepatic vein.

1.4 Contents and organization

The remainder of this dissertation is organized as follows. First, related works are explored in chapter 2. In chapter 2, an introduction of CNN, several modern architectures of CNNs, and state-of-the-art models for medical image segmentation are illustrated. Subsequently, the literature on liver and vessel segmentation is presented. An active contour model-based segmentation technique is thoroughly reviewed, and finally, an active contour model that employs a topology of a vascular structure is introduced. The proposed methods for the liver and its vessel segmentation are described in chapters 3 and 4, respectively. Each chapter comprises an overview, detailed methodology, corresponding experimental results, and discussion. The conclusion and future works are presented in chapter 5.

Chapter 2

Related Works

2.1 Overview

In this chapter, a literature review regarding liver and vessel segmentation is illustrated. The chapter is composed of the two main subjects: 1) convolutional neural networks for medical image segmentation and 2) literature of liver and vessel segmentation. First, convolutional neural networks are introduced and networks that are related to medical image segmentation are highlighted. The presented networks are used in the later experimental assessments. The following sections demonstrate a literature review of algorithms for liver and vessel segmentation. Especially, an active contour model, which is closely related to the proposed method in this dissertation, is thoroughly reviewed. Finally, common limitations of the current algorithms and the corresponding motivations are illustrated in the final motivation section.

2.2 Convolutional neural networks

In this section, an introduction to the convolutional neural network (CNN), several CNN architectures, and CNNs for medical image segmentation are featured. A brief architecture of CNNs and building blocks are firstly reviewed and several state-of-the-art CNNs are highlighted that performs medical image segmentation tasks.

2.2.1 Architectures of convolutional neural networks

Brief introduction to neural networks

An artificial neural network is designed in the spirit of mimicking the human brain's neuron activation. A multi-layer perceptron [12, 13] has been a basic architecture in a feed-forward fashion [12]. The input signal is fed to the network and propagated forward by weight parameters to output a new signal. Intermediate signals are fully connected (FC) which can be represented by a matrix multiplication formed with trainable parameters (Fig. 2.1):

$$\mathbf{h}_l^m = W_l^{m \times n} \mathbf{h}_{l-1}^n + \mathbf{b}_l, \tag{2.1}$$

where $\mathbf{h}$, W, and $\mathbf{b}$ represents a feature of hidden layer (i.e., intermediate signal), weight matrix, and bias vector, respectively. The variables m, n, and W_l denote dimension of l^{th}, $l-1^{th}$ hidden layer, and the inter FC weights (i.e., matrix), respectively. To employ non-linear transformation which can distinguish data that is not linearly separable, non-linear activation functions are applied to every output of FC layers. The layers are stacked deeply and trained so that the architecture is called "deep learning" or "deep neural network" in our modern academic societies. A neural network is trained by designing a differentiable loss function at the final output layer. The calculated loss is then

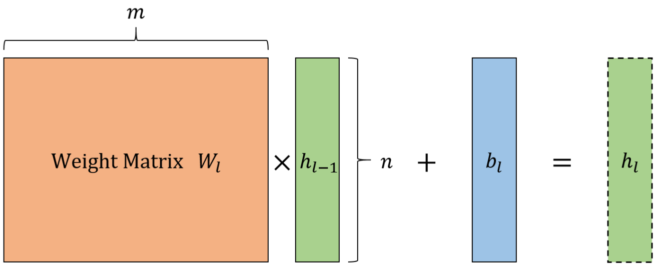


Figure 2.1: An example fully connected layer formed by matrix multiplication. A weight matrix (W_l) and biases (b_l) are trainable parameters.

back-propagated [14] to intermediate neurons via partial derivatives:

$$\frac{\partial L}{\partial W}, \tag{2.2}$$

where L indicates the objective (i.e., loss) function and W denotes a set of all trainable parameters.

The main significance of neural network architecture is that features, which were traditionally human-designed, are extracted automatically when a certain task and the corresponding data are given. That is, a neural network is a data-driven algorithm that learns from data referring to tasks. In earlier days, the computational power could not afford the large neural architectures to be employed in real-time applications. Furthermore, the amount of available data has been limited in various fields. In recent years, deep learning has been gaining its capability and popularity owing to the increase of training data and the improvement of hardware infrastructures.

The impact of deep neural networks has been groundbreaking in many applications. The main factor of success is a tremendous gain of the complexity of feature extraction and description that are automatically trained from a large amount of data. Deeply stacked layers act as a feature transformation which transforms the feature space to be well-disentangled (i.e. representative) that

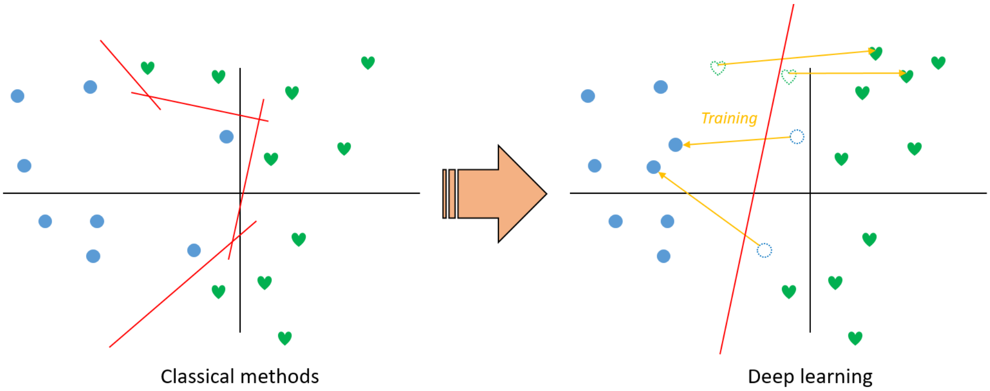


Figure 2.2: Difference between classical methods and deep learning. Classical methods attempt to extract discriminable features in a human-designed manner and optimize classifiers for given tasks. On the other hand, a deep learning-based approach aims to learn optimal feature extractors that can be discriminable so that the final classification can be easier.

the classes are to be effectively classified for the final decisions (Fig. 2.2). The application of deep neural networks indicates the construction of a classifier with intractable complexity. It is indeed obtained from a huge number of parameters formed by many neural layers [15,16]. However, deep learning is nowadays facing a great challenge in improving generalization [17]. A foundation of deep learning is a data-driven method that lies under almost every modern architectures. The problem arises from the training data in perspective of its amount and distribution. The algorithm is prone to be over-fitted to the training data primarily due to the large complexity of networks. Until recently, many applications suffer to employ deep learning because of data deficiency and imbalance. Oppose to a huge trend of neural architecture search which attempts to automate the entire training procedure from the neural architecture designing step, it is still critical to build a human-designed algorithm with domain-specific knowledge for the

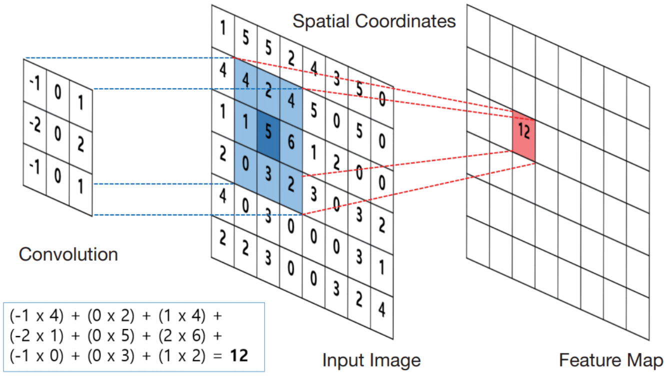

Spatial Coordinates
Convolution
(-1 x 4) + (0 x 2) + (1 x 4) +
(-2 x 1) + (0 x 5) + (2 x 6) +
(-1 x 0) + (0 x 3) + (1 x 2) = 12
Input Image
Feature Map
12

Figure 2.3: The convolution operation. The weights of a convolutional kernel are weighted summed in a sliding window fashion for the input image (or intermediate features in the neural networks). Weights of a convolutional kernel are trainable parameters in convolutional neural networks.

better generalization performance to empower deep neural networks.

Convolution operator

The basic building operations of a convolutional neural network are convolution (Fig. 2.3) and non-linear activation functions. The feature map after convolution, including the input image, typically passes through a non-linear activation function to obtain non-linear combination of signals. A non-linear layer (i.e., feature transformation) typically comprises a composite operations such as convolution, batch normalization [18], and non-linear unit (e.g., rectified linear unit [19] non-linearity):

$$\mathbf{f}_l = F_l(\mathbf{f}_{l-1}) = \sigma(b((\mathbf{f}_{l-1} * \theta_l), \gamma_l)), \tag{2.3}$$

where θ_l is a trainable weights of a convolutional kernel in the i^{th} layer, $\mathbf{f}_i$ is the i^{th} layer features, $b(\mathbf{f}, \gamma)$ is a batch normalization [18] which transforms

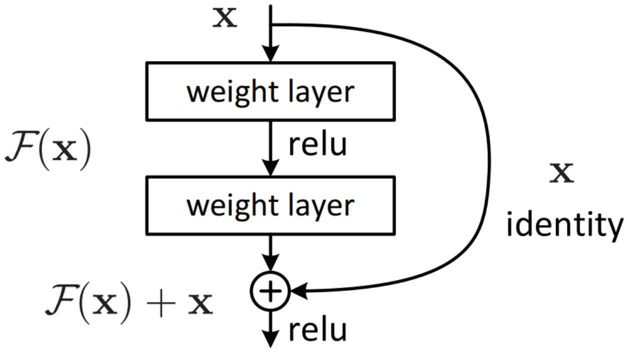


Figure 2.4: A building block of ResNet. The input signal is added to the output (i.e., skip connected).

the mean and variance of each channel to 0 and γ (trainable scale parameter), and σ indicates a non-linear function. Non-linear layers in this dissertation are composed of the three operations as defined in (2.3) unless some operations are specially omitted. Note that the literature in this section is not restricted to the CNN architecture.

Residual connections

A neural network with residual connections (ResNet) has been proposed for image classification [20]. A building block of ResNet is an identity shortcut connections which add an input signal directly to the output of non-linearities (Fig. 2.4):

$$\mathbf{f}_l = F_l(\mathbf{f}_{l-1}, W_l) + \mathbf{f}_{l-1}, \tag{2.4}$$

where W_l is a set of weights correspondingly associated with the l^{th} layer. That is, the skip connections between layers add the outputs from previous layers to the outputs of stacked layers. Therefore, derivative of a certain layer $\mathbf{f}_l (1 \leq l \leq L)$ can be represented according to the chain rule of back-propagation

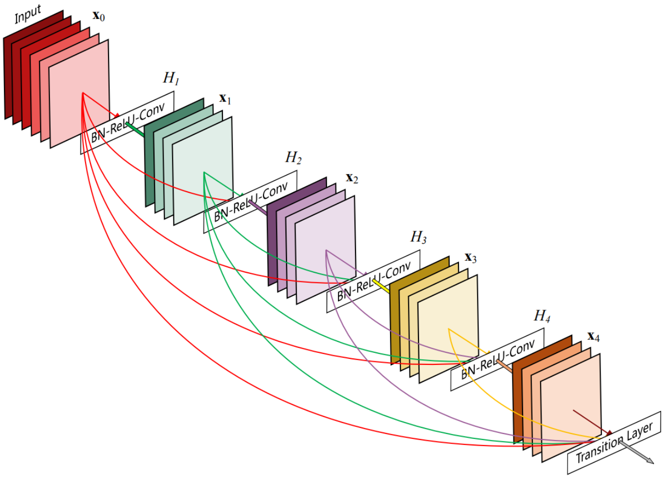


Figure 2.5: Densely connected convolutional network [21].

[14]:

$$\frac{\partial L}{\partial \mathbf{f}_l} = \frac{\partial L}{\partial \mathbf{f}_L}\frac{\partial \mathbf{f}_L}{\partial \mathbf{f}_l} = \frac{\partial L}{\partial \mathbf{f}_L}(1 + \frac{\partial}{\partial \mathbf{f}_l}\sum_{i=l}^{L} F_i(\mathbf{f}_i, W_i)), \tag{2.5}$$

where L denotes the loss function of deep residual networks.

The architecture of skip connections via residual shortcut made it possible to train much deeper networks [20]. The gradients of a loss function flow through skip connections so that the gradient vanishing problem has been eased.

Densely connected convolutional network

Densely connected convolutional network (DenseNet) [21] connects each layer to every other layer in a feed-forward fashion (Fig. 2.5). The main advantage of the presented architecture is that the gradient directly flows to deep layers, accelerating the learning procedure. Feature reusing scheme also strongly contributes to a substantial reduction in the number of parameters. This structure can be viewed as an implicit deep supervision network similar to the explicit version [22]. The l^{th} layer obtains the concatenation of all outputs of the preceding layers [21]:

$$\mathbf{f}_l = F_l([\mathbf{f}_0, \mathbf{f}_1, ..., \mathbf{f}_{l-1}]), \tag{2.6}$$

where $\mathbf{f_l}$ denotes the output of the l^{th} layer and $[\mathbf{f}_0, \mathbf{f}_1, ..., \mathbf{f}_{l-1}]$ refers to a concatenation of feature-maps produced in the previous layers. A feature-reusing scheme of DenseNet, which causes a reduction of parameters, is an effective feature for the 3D volumetric neural network because volumetric data easily lack GPU memories due to deep stacks of layers in DNNs.

Depth-wise separable convolutions

Depth-wise separable convolutions [23] showed ground-breaking results with a separation of features in a depth-wise (i.e., channel-wise) manner. Separable convolution is performed in depth-wise channel separation and further concatenation:

$$\ddot{F}(\mathbf{x}) = \sigma(b([\mathbf{f}_{0..k-1} * \theta_0, ..., \mathbf{f}_{c-k..c-1} * \theta_{c-1}], \gamma)), \tag{2.7}$$

where c indicates the number of channels of feature $\mathbf{f}$ and k denotes the number of channels of each separated group. An application of separable convolutions showed better accuracy with a simple structure compared to Inception V3 [24] module which is formed by a complex composite of bottleneck layers. Furthermore, the effective use of parameters improved generalization performance.

Deeply supervised networks

A deep supervision metric [22] was proposed by introducing classifiers at hidden (i.e., intermediate) layers. The key underlying concept of deep supervision metric is that a discriminative classifier that is trained on highly discriminative features are more likely to derive better performance than a discriminative classifier trained on less discriminative features [22]. The method also alleviates

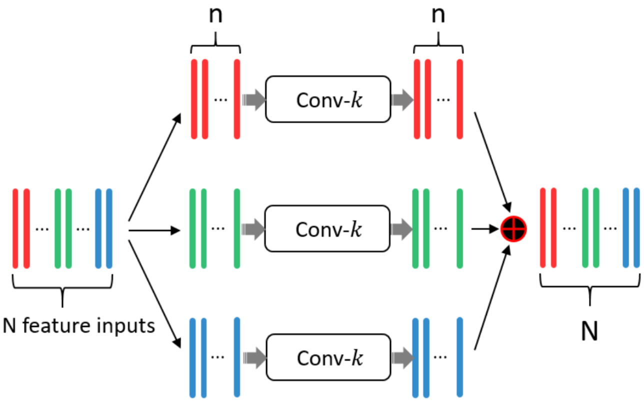


Figure 2.6: An example of depth-wise separable convolutions. k indicates the size of a kernel. The channels are separated and pass through a convolution for the final output features which are obtained by simple concatenation.

the gradient vanishing problem. A deeply supervised network showed a new possibility of auxiliary classifiers that can be adapted to multi-task neural networks [10, 25]. Originally, a deeply supervising method was proposed to add a loss to the intermediate layers to enhance the discriminability of the low-level features. The method was also proved to achieve improved performance of generalization. The spirit of deep supervision was successfully applied to a liver segmentation [26] which are reviewed in section 2.2.2.

Attention mechanism

The attention mechanism is literally "paying more attention" to certain intermediate features (or gradients) to improve the performance of neural networks [27]. The primary idea of attention mechanism is applied by generating an attention vector that assigns relative weights on a sequence of features. In the application of natural language processing, the attention mechanism showed groundbreaking results in the field such as machine translation [27–31] and classification [32–34]. The superior feature of attention modules is its capability of modeling long-range dependencies [30].

There are many studies that relate the attention mechanism with computer vision tasks such as image classification [33–36], segmentation [37–41], detection [42], action recognition [43–45], image captioning [46, 47], visual question answering [48, 49], and pose estimation [50]. The primary goals of employing attention mechanisms to the field of computer vision are to increase the neural network's discriminability and to effectively incorporate local and global features. The attention mechanism typically enhances neural networks to focus on the most relevant (i.e., important) features without additional deep supervision metric which has been a prominent method to make intermediate features representative [22]. The aspect of highlighting salient features and avoiding the use of multiple redundant features, attention mechanisms greatly contribute to the compactness and discriminability of neural networks. The compactness of the network is typically achieved by the self-attention [34, 45] method which does not use the external information. For example, a non-local self-attention was used to capture long-range dependencies [45] and a class-specific pooling was performed via self-attention [33, 34]. That is, an attention is applied to weight self-features (i.e., internal features for each layer or module) which are to be critical to a given task. There are several works that employs channel-wise attention [51], spatial-wise attention [52], or both [41] into the neural networks. Channel-wise attention [51] gives class-wise (i.e., feature-wise) attention to weigh relative importance among features. Spatial-attention, on the other hand, applies attention in a spatial manner to make layers to focus on certain spatial regions [41].

A multi-scale analysis of images has been a great success in computer vision tasks [40, 41]. Low-level features focus on local appearance while higher-level features encode global representations. The attention mechanism can be greatly incorporated with neural networks to delineate the optimal combination between local and global features. The self-supervision metric has been successfully employed to model the integration of local and global relationships [40,41].

In recent years, the attention mechanism has been adapted to medical image segmentation tasks [52–55]. In [53], multi-resolution features were successfully combined by integrating local deep attention features and a global context. More recently, attention gated networks [52] has been proposed to leverage low- and high-level features via attention gates. Grid-based attention was employed to allow attention gates to be more specific to local regions [52]. A guided attention [55] attempted to incorporate multi-scaled features using intermediate layers of ResNet [20]. The authors have employed both position and channel attention modules and incorporated guided loss by encoder and decoder networks [55].

Penalizing confident output distributions

A method of regularizing the neural networks by penalizing the confident output distribution has been proposed in [8]. Different from manually manipulating the training distributions [56], the authors have analyzed the output of the network [8]. The low entropy of the output distribution was defined as a confident output and the over-confident symptom as an over-fitting [24]. The proposed confident penalty constitutes a regularization term that prevents peak distribution which leads to a better generalization [8].

A conditional distribution of a neural network's output (after softmax) can be defined as $p_W(\mathbf{y}|\mathbf{x})$, where $\mathbf{x}$ is an input, $\mathbf{y}$ is a class vector, and W is a set of parameters. The entropy of this conditional distribution is given by

$$H(p_W(\mathbf{y}|\mathbf{x})) = -\sum_i p_W(\mathbf{y_i}|\mathbf{x})\log(p_W(\mathbf{y_i}|\mathbf{x})). \tag{2.8}$$

A simple penalization of the confident output can be achieved by adding the negative entropy to the negative log-likelihood during training:

$$L(W) = -\sum \log(p_W(\mathbf{y}|\mathbf{x})) \underbrace{-\beta H(p_W(\mathbf{y}|\mathbf{x}))}_{\text{negative entropy}}, \tag{2.9}$$

where β is a control parameter for the strength of the penalty. Another penalization for the supervised learning, which desires to be converged fast, can be designed as [8]:

$$L(W) = -\sum \log(p_W(\mathbf{y}|\mathbf{x})) - \beta \underbrace{\max(0, \rho - H(p_W(\mathbf{y}|\mathbf{x})))}_{\text{hinge loss}}, \tag{2.10}$$

which penalizes the output distribution when they are below a certain entropy threshold, ρ.

The penalizing the confident output distribution [8] and the attention mechanism [33, 34, 51, 57] are similar in the perspective of internally weighting the neural network to boost the accuracy. An attention method is applied to intermediate layers that weigh the feature maps by either channel- or spatial-wise manner [51, 57]. On the other hand, penalizing the output method attempts to modify the final loss function to regularize the network.

2.2.2 Convolutional neural networks in medical image segmentation

3D U-Net

3D U-Net [58] extended the U-Net [59] architecture by replacing all the 2D operations with their 3D counterparts. Volumetric 3D convolutions improved the network by extracting 3D contextual information. A review of the U-Net architecture is presented in the following paragraph.

In recent years, the U-Net [59] has been the most popular neural network which was adapted and improved by a huge body of literature. U-Net is one

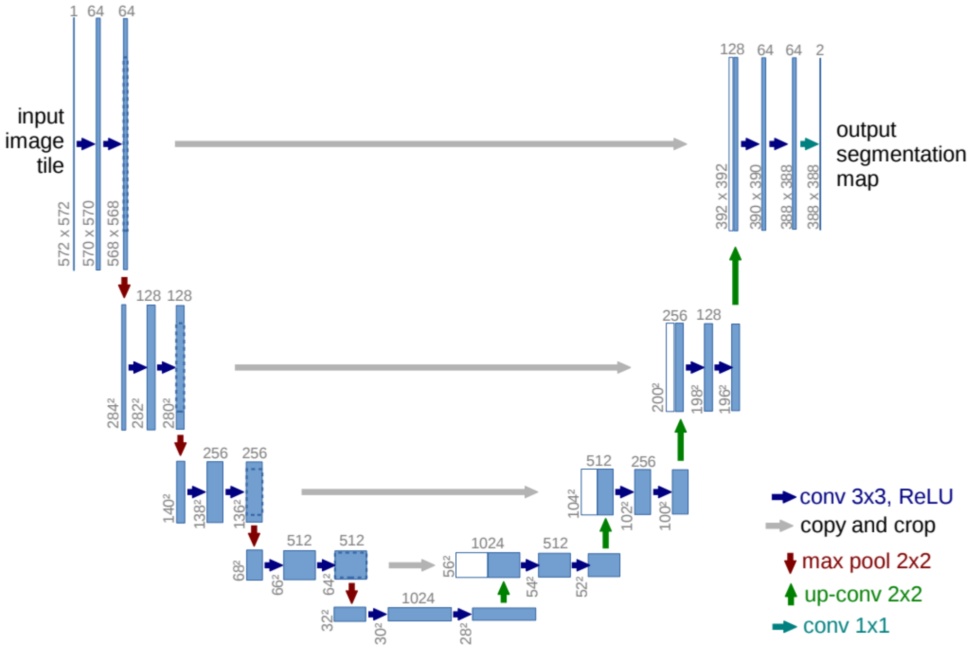


Figure 2.7: U-Net architecture [59]. Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the boxes. The dimensions are provided at the lower-left edge of the boxes. White boxes represent copied feature maps that are concatenated.

of the main architecture of modern neural networks that are related to imaging applications. The main underlying principle of U-Net is a combination of low- and high-level features in a fully convolutional fashion. As the title gives a rough intuition of the architecture, ‘U’ represents the shape of the proposed network (Fig. 2.7). The network includes a contracting (left side) and an expanding (right side) paths of deep intermediate features. The contracting path is composed of 3^2 convolution layers followed by 2^2 max pooling. The number of feature maps (i.e., the number of channels) is doubled after contraction. A 2^2 up-convolution layer is used in the expanding paths. For the up-convolutional features in the expanding paths, the number of features was halved and features from the contracting paths at the same level are concatenated for further propagation. The skip connections (i.e., concatenations) of the two features are the key component of the U-Net. The combined features can be jointly convolved to extract multi-scaled features that have low- and high-level representations.

Training of the network was performed by a pixel-wise softmax over the final feature map combined with the cross-entropy loss function [59]:

$$L_{unet} = -\sum_{\mathbf{x}\in\Omega} w(\mathbf{x})\log\big(p_{\mathbf{y}(\mathbf{x})}(\mathbf{x})\big), \tag{2.11}$$

where Ω denotes the image dimensions and $p_{\mathbf{y}(\mathbf{x})}(\mathbf{x})$ is the softmax defined as

$$p_k(\mathbf{x}) = \frac{\exp(a_k(\mathbf{x}))}{\sum_j \exp(a_j(\mathbf{x}))}, \tag{2.12}$$

where $a_k(\mathbf{x})$ denotes the k^{th} channel at the final activation layer and j denotes a channel index. $p_k(\mathbf{x})$ is the approximated maximum function that is $p_k(\mathbf{x}) = 1$ for the k that has the maximum activation $a_k(\mathbf{x})$ and $p_k(\mathbf{x}) = 0$, otherwise. $\mathbf{y}$ is the ground-truth label of each pixel and $w(\mathbf{x})$ is a pre-computed weighting map for each ground-truth segmentation to compensate the different frequency of pixels from a certain class in the training data [59]. Note that a softmax with weighted cross-entorpy loss was used in 3D U-Net [58] different from the original U-Net objective function (2.11) [59]:

$$L_{3dunet} = -\sum_{c} \mathbf{y}_c\log\tilde{\mathbf{y}}_c + \lambda||W||_2^2, \tag{2.13}$$

where $\tilde{\mathbf{y}}$ denotes the predicted probability of class c after softmax operation and $\mathbf{y} \in \{0, 1\}$ is the corresponding ground-truth (i.e., $\mathbf{y}_{c,i} = 1$ if voxel i belongs to the class c, otherwise 0).

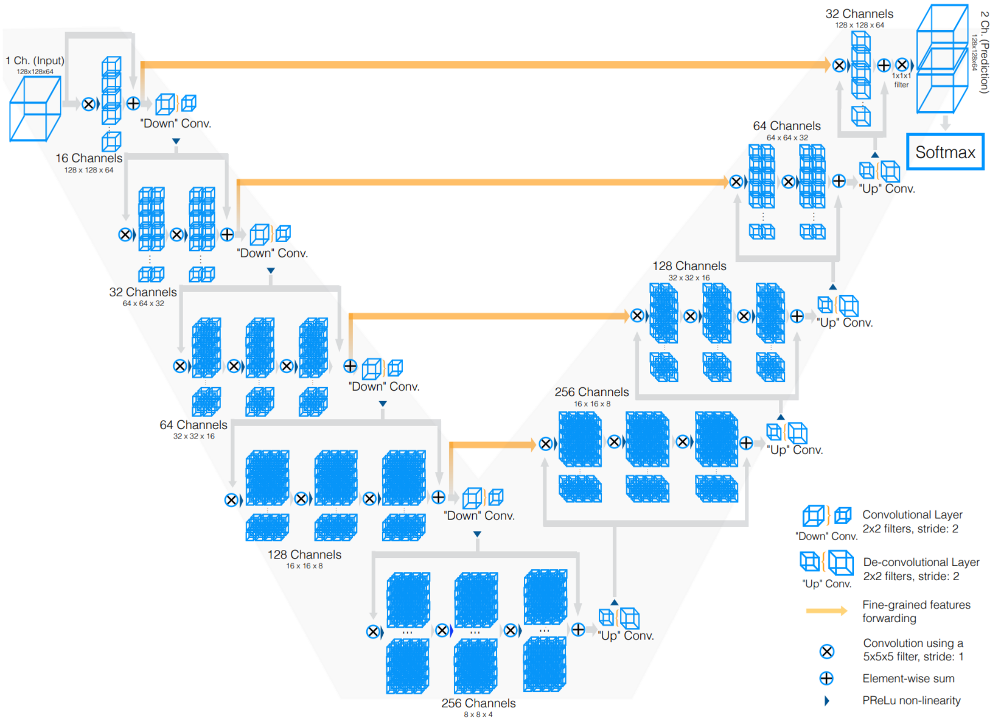


Figure 2.8: V-Net architecture [60].

V-Net

V-Net [60] is a volumetric FCN for medical image segmentation. U-Net architecture [59] was extended to volumetric convolution (i.e., 3D convolution) and U-Net-like downward and upward transitions (i.e., convolutional reduction and de-convolutional expanding of feature dimensions; for more details, refer to the original work [60]) were adopted together with many skip connections via an element-wise summation scheme (Fig. 2.8). The main significant difference from the standard U-Net [59] is the employment of multiple residuals [60]. The dice loss was first presented in the application of image segmentation to overcome the class imbalance problem [60]:

$$D = \frac{2\sum_i^N p_i g_i}{\sum_i^N {p_i}^2 + \sum_i^N {g_i}^2}, \tag{2.14}$$

where p_i and g_i are the binary voxels in each predicted binary and the ground-truth volume. The dice formulation can be differentiated yielding the gradient

$$\frac{\partial D}{\partial p_j} = 2 \times \left[\frac{g_j(\sum_i^N {p_i}^2 + \sum_i^N {g_i}^2) - 2p_j(\sum_i^N p_i g_i)}{(\sum_i^N {p_i}^2 + \sum_i^N {g_i}^2)^2} \right] \tag{2.15}$$

computed with respect to the j^{th} voxel of the prediction [60]. The final loss function of V-Net training is as follows:

$$L_{vnet} = D(\tilde{\mathbf{y}}_{0.5}, \mathbf{y}) + \alpha||W||_2^2, \tag{2.16}$$

where $\tilde{\mathbf{y}}_{0.5}$ is the binary output prediction thresholded by 0.5 after softmax, $\mathbf{y}$ indicates the ground-truth label image, W denotes the set of parameters of the network, and α is a weighting coefficient.

The main significance of V-Net architecture is the introduction of the dice loss (2.14) and a fully convolutional volumetric neural network for medical image segmentation. The dice loss intrinsically overcomes the class imbalance problem by avoiding strong bias towards background learning. The dice loss does not require weighting parameters for the loss function to assign proper weights to samples of different classes to establish the right balance between foreground and background voxels [60]. The latter fully convolutional volumetric architecture showed a promising direction to employ end-to-end learning framework. In [60], all the training volumes have resized to a 128 × 128 × 64 grid of voxels and trained in an end-to-end manner.

Deeply supervised network

Deeply supervised network (DSN) [26] has been proposed to supervise a network in a deep-level. The network was designed in the spirit of deep supervision

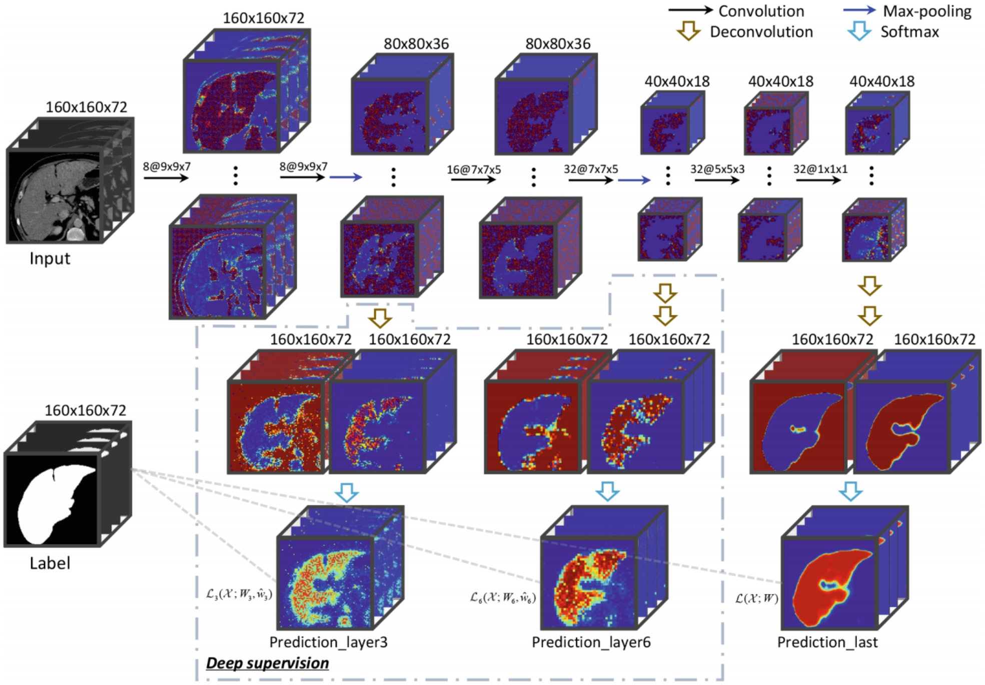


Figure 2.9: Deeply supervised network for liver segmentation [26]. The architecture of the proposed 3D DSN deeply supervises intermediate feature volumes and predicts the score at the last layer.

[22] which enhances the discriminability of the low-level features so that the final classifier can easily be a better discriminative classifier which results in the improvement of the final accuracy. Accordingly, a loss function penetrates through multiple layers in a DNN (Fig. 2.9). Another aspect of DSN is that training difficulty owing to exploding and vanishing gradient problems can be alleviated by direct and deep gradient flows. In [26], a 3D deep supervision mechanism has been adapted to volumetric medical image segmentation. The authors exploited two explicit deep supervisions to hidden layers and those auxiliary losses were integrated to the final loss with the last output layer

to back-propagate the gradients [26]. Each intermediate layer has a different resolution (via max-pooling [61]) to improve the multi-scaled features that can be representatively integrated. The overall loss function of DSN is as follows:

$$L_{dsn} = \sum_{i} -\text{log}p(\mathbf{y}_i|\mathbf{x}_i; W) + \sum_{d \in D} \eta_d L_d(\mathbf{x}, \mathbf{y}; W_d, \hat{w}_d) + \lambda(||W||^2 + \sum_{d \in D} ||\hat{w}_d||^2), \tag{2.17}$$

where $\mathbf{x}_i$ is the i^{th} input image, $\mathbf{y}_i$ is the i^{th} ground-truth label, W is a set of all parameters, and L_d represents the auxiliary losses (i.e., deep supervisions) defined as

$$L_d(\mathbf{x}, \mathbf{y}; W_d, \hat{w}_d) = \sum_{i} -\text{log}p(\mathbf{y}_i|\mathbf{x}_i; W_d, \hat{w}_d), \tag{2.18}$$

where W_d denotes the set of parameters before the d^{th} auxiliary classification and D is the set of indices of all the hidden layers which are equipped with the deep supervision [26]. $\hat{w}_d$ represents the weights which bridge the d^{th} auxiliary layer feature volumes to dense predictions (i.e., de-convolution layers in the original paper [26]). The parameters η_d and λ are the balancing weights of the overall objective function. The negative log-likelihoods were applied to calculated the probability $p(\mathbf{y}|\mathbf{x})$ in [26].

The authors of DSN [26] additionally applied a classical post-refinement step via employing a conditional random field (CRF) for contour refinement. To overcome misclassified regions especially in ambiguous boundaries, the posterior of a liver (i.e., the output of the network) was jointly combined with the original image to model an energy function [26]:

$$E(\mathbf{y}) = \underbrace{\sum_{i} -\text{log}\hat{p}(\mathbf{y}_i|\mathbf{x}_i)}_{\text{unary potential}} + \underbrace{\sum_{i,j} f(\mathbf{y}_i, \mathbf{y}_j)\phi(\mathbf{x}_i, \mathbf{x}_j)}_{\text{pairwise potential}}, \tag{2.19}$$

where the first term is the unary potential indicating the distribution over label assignment y_i at a voxel x_i. To aggregate multi-scale information, the $\hat{p}(y_i|x_i)$ is initialized as a linear combination of the last output layer and the intermediate predictions (i.e., deeply supervised layers) obtained from the network:

$$\hat{p}(\mathbf{y}_i|\mathbf{x}_i) = \Big(1 - \sum_{d \in D} \tau_d\Big) p(\mathbf{y}_i|\mathbf{x}_i; W) + \sum_{d \in D} \tau_d p(\mathbf{y}_i|\mathbf{x}_i; W_d, \tilde{w}_d). \tag{2.20}$$

The second term in (2.19) is the pairwise potential, where $f(y_i, y_j) = 1$ if $y_i \neq y_j$, and 0 otherwise [26]. The $\phi(x_i, x_j)$ incorporates the local appearance and smoothness by employing the gray-scale value I_i and I_j and bilateral position s_i and s_j of the voxel x_i and x_j as follows:

$$\phi(\mathbf{x}_i, \mathbf{x}_j) = \mu_1 \mathrm{exp}\Big(-\frac{||\mathbf{s}_i - \mathbf{s}_j||^2}{2{\theta_\alpha}^2} - \frac{||I_i - I_j||^2}{2{\theta_\beta}^2}\Big) + \mu_2 \mathrm{exp}\Big(-\frac{||\mathbf{s}_i - \mathbf{s}_j||^2}{2{\theta_\gamma}^2}\Big). \tag{2.21}$$

The constant weights τ_d in the unary potential (2.20) and parameters μ_i, θ_α, θ_β, and θ_γ in the pairwise potential (2.21) were optimized using a grid search on the training set [26].

Voxel-wise residual network

Voxel-wise residual network (VoxResNet) was proposed to resolve brain segmentation task [62]. The base module of the network is voxel-wise residual unit (Fig. 2.10). The module comprises a series of convolution, batch normalization [18], and a rectified linear unit (ReLU) non-linearity [19]. The input was skip connected in a residual manner (2.4).

VoxRes modules are deeply stacked with several convolutional layers and deep, multi-scaled supervisions (Fig. 2.10). The full architecture of VoxResNet is similar to that of DSN [26] from the perspective of deep supervisions. Oppose to the DSN [26] method, which deeply supervised intermediate layers individually, the main difference of the base VoxResNet architecture is that the intermediate layers and the final prediction are summed over to apply overall loss function:

$$L_{voxresnet} = -\sum_c \mathbf{y}_c \mathrm{log}\tilde{\mathbf{y}}_c - \sum_a \sum_c w_a \mathbf{y}_c \mathrm{log}{\tilde{\mathbf{y}}_c}^a + \lambda ||W||_2^2, \tag{2.22}$$

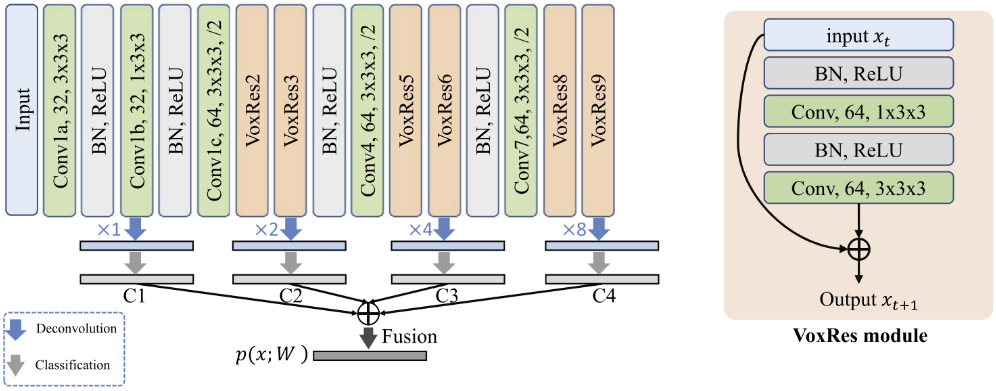


Figure 2.10: Voxel-wise residual network [62]. Voxel-wise residual modules are deeply stacked in the network.

where $\tilde{\mathbf{y}}$ denotes the predicted probability of class c after softmax classification layer and $\mathbf{y}_c \in \{0, 1\}$ is the corresponding ground-truth (i.e., $\mathbf{y}_{c,i} = 1$ if voxel i belongs to the class c, otherwise 0). The w_a, where a indicates the index of auxiliary classifiers, is the weights of auxiliary classifiers [62].

The authors extended their work by employing an auto-context framework using VoxResNet as a baseline [62]. An auto-context version of the VoxResNet was proposed by combining the low-level image appearance features, implicit shape information, and high-level context together for further improving the segmentation performance [62]. The two identical VoxResNet networks were used for the posterior and the final inference of an auto-context version (Fig. 2.11). The authors first trained a VoxResNet classifier on the original training sub-volumes with image appearance information (i.e., the original volumes) [62]. Then, the output of the first VoxResNet is used as a context information at a higher level, discriminative probability maps. The original volumes (i.e., appearance information), together with a context information were concatenated as another input to train a new classifier (i.e., the second VoxResNet) [62].

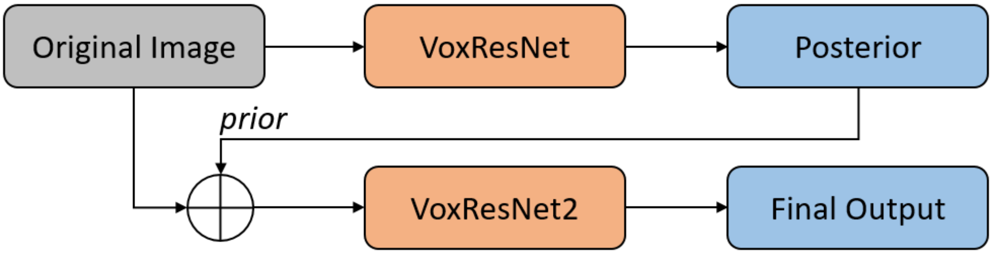


Figure 2.11: Architecture of voxel-wise residual network (VoxResNet). The two identical VoxResNet is applied to acquire prior (i.e., posterior of the first VoxResNet) and the final output. Two identical networks are required to be trained for two-step inference.

Dense V-Networks

A simultaneous multi-organ segmentation on abdominal CT images has been proposed [63]. A densely connected convolutional layers [21] were employed as a unit block for the network (Fig. 2.12). A dense block units are structured as a V-Net-like structure [60] (DenseVNet) to extract low- and high-level features, and further fused (i.e., summed) for the final output [63]. Spatial-wise dropouts were employed to the proposed dense blocks [63].

The authors introduced a new dice objective function that mitigates the extreme class imbalance [63]:

$$D'(\tilde{\mathbf{y}}_l, \mathbf{y}_l) = \overline{\left(\frac{min(\tilde{\mathbf{y}}_l, 0.9) \cdot \mathbf{y}_l}{||\mathbf{y}_l||_2 + ||min(\tilde{\mathbf{y}}_l, 0.9)||_2}\right)}, \tag{2.23}$$

where $\tilde{\mathbf{y}}_l$ is the softmax output of the network which is a probabilistic segmentation and $\mathbf{y}_l$ is the binary ground-truth label for organ l for each subject. The dice scores for each organ l were averaged across subjects in each minibatch [63]. Dice score hinge losses heavily penalizing dice scores below 0.001 and 0.10 were introduced after warm-up periods of 25 and 100 iterations, respectively. Thus

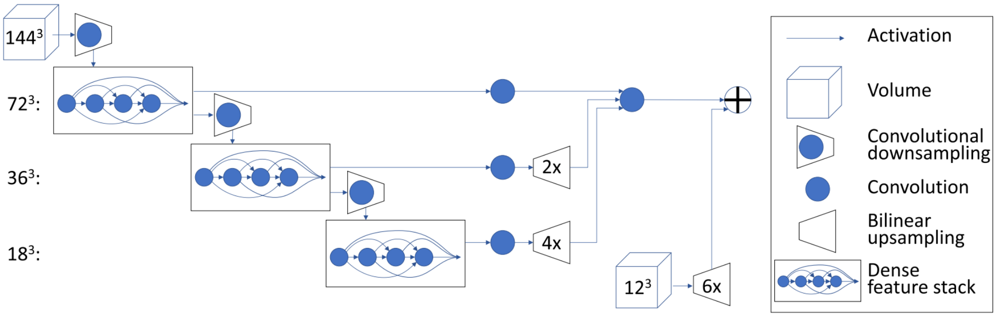


Figure 2.12: Architecture of dense V-Networks (DenseVNet) [63]. Densely connected blocks are schematically visualized. 12^3 sized trainable grid is employed to train a shape prior. The shape prior is added to the final prediction.

the loss function at i^{th} iteration was defined as follows:

$$L_{densevnet}(\tilde{\mathbf{y}}, i) = -\frac{1}{8}\sum_{l} d(D'(\tilde{\mathbf{y}}_l, \mathbf{y}_l), i) + \sum_{w \in W} \frac{\overline{w^2}}{40}, \tag{2.24}$$

where d is defined as

$$d(D', i) = D' + 100h(D', i, 0.01, 25) + 10h(D', i, 0.1, 100), \text{and} \tag{2.25}$$

$$h(D', i, v, t) = sigmoid(6(i-t)/t)max(0, v - D')/v^4, \tag{2.26}$$

where v is the hinge loss threshold, and t is the delay in iterations [63].

A singularity of DenseVNet is that the trainable grid was introduced to learn the shape prior. The authors argued that medical images are frequently acquired in standard anatomically aligned views with relatively consistent organ positions and orientations [63]. In assuming the spatial data coherency, the network used an explicit spatial prior which are trainable grid-parameters representing a prior shape probability. The resolution of the spatial prior was 12^3 which was up-sampled by the factor of 6 resulted in the final output resolution 72^3. The prior was added to the final output prediction layer.

Attention gated U-Net

An attention mechanism has been incorporated into the medical image segmentation networks in [52,64]. The authors proposed an attention gated mechanism which can be easily integrated into standard CNN architectures. The proposed attention gating method was employed to a standard U-Net (AGU-Net) [59]. An adaptive feature pooling that allows attention to be performed on specific local regions was successfully applied via an image-grid based gating mechanism. Rather than using a global vector for all image pixels, the grid signal which is conditioned to image spatial information was employed. The primary achievements by designing an attention gating mechanism in [52] were to replace external organ localization models and eliminate the need for labeled box annotations, and remove back-propagation-based saliency map generation. The authors argue that the segmentation tasks were successfully performed without a conventional cascading localization framework or two-stage recurrences [65–68].

An attention gate (AG) module was developed to disambiguate task-irrelevant feature contexts in the intermediate layers. AGs were applied to every skip connection right before the concatenation on a standard U-Net which were to jointly attend the features at multi-scales. In other words, every skip connected feature passes through an AG module which applies attention using the input features and the higher-level features from contracted paths (Fig. 2.13). The features on the higher level, which have the coarse spatial dimensions, were used as gates that were intended to make AG modules (Fig. 2.14) better to jointly attend the local- and global-scaled features to rule out irrelevant background regions. Let ${\mathbf{f_i}}^s$ be the feature map at scale s ($s \in \{1..3\}$) which is an index value of l^{th} layer ($l \in \{1..L\}$). Each ${\mathbf{f_i}}^s$ indicates the output features for skip connections from the contracting paths that pass through an attention gate. The higher

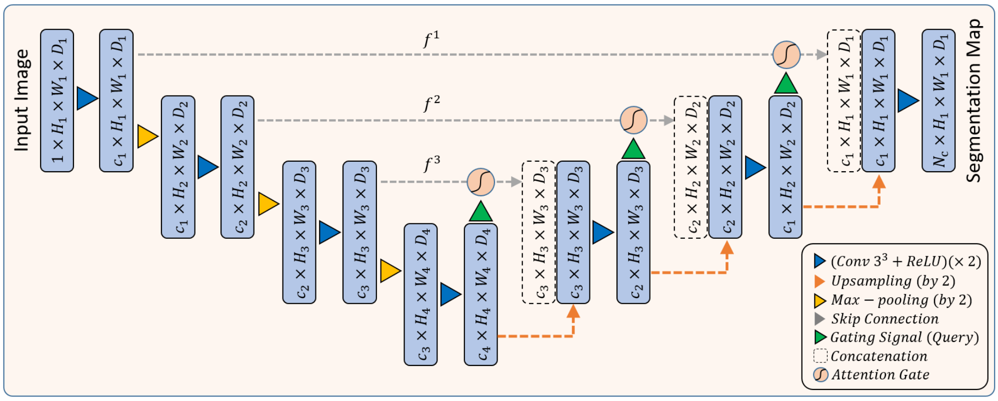


Figure 2.13: Attention gated U-Net architecture [52]. An input image is progressively filtered and down-sampled by the factor of 2 at each scale (s) in the encoding part of the network (i.e., contracting paths). Attention gates (AGs) weigh the features that are propagated through the skip connections right before the concatenation.

scale indicates the more contracted features that are in the coarser spatial dimensions. $\mathbf{f_i}^s$ represents a pixel-wise feature vector, and the length is defined by the number of channels at a given scaled layer (i.e., $\mathbf{f_i}^s \in \mathbb{R}^{c_s}$). For each $\mathbf{f_i}^s$, AG computes a coefficient map α_i^s, where $\alpha_i^s \in [0, 1]$. The coefficient map attends salient image regions and prunes false responses to the background. The output of AG module is defined as $\hat{\mathbf{f}}_{\mathbf{i}}^s = \{\alpha_i^s \mathbf{f_i}^s\}_{i=1}^n$, where n denotes the size of each feature [52]. That is, the coefficients α_i^s act as spatial attention for each feature vector. The authors used additive attention [28, 69] rather than multiplicative attention [29]. The aforementioned attention coefficients can be formulated as follows [52]:

$$\alpha_i^s = \sigma_2\Big(\psi^T\big(\sigma_1(\mathbf{W}_f^T \mathbf{f_i}^s + \mathbf{W}_g^T \mathbf{g^{s+1}} + \mathbf{b}_{fg})\big) + \mathbf{b}_\psi\Big), \tag{2.27}$$

where σ_1 is an element-wise ReLU (i.e., $\sigma_1(x_{i,c}) = max(0, x_{i,c})$) and σ_2 is a

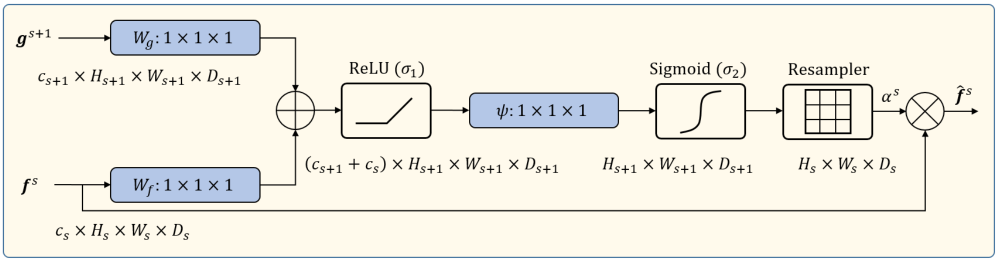

g^{s+1}
$W_g: 1\times1\times1$
$c_{s+1}\times H_{s+1}\times W_{s+1}\times D_{s+1}$
ReLU (σ_1)
$\psi: 1\times1\times1$
Sigmoid (σ_2)
Resampler
α^s
$\hat{f}^s$
$(c_{s+1}+c_s)\times H_{s+1}\times W_{s+1}\times D_{s+1}$
$H_{s+1}\times W_{s+1}\times D_{s+1}$
$H_s\times W_s\times D_s$
f^s
$W_f: 1\times1\times1$
$c_s\times H_s\times W_s\times D_s$

Figure 2.14: Attention gate module proposed in [52]. Input features ($\mathbf{f}^s$) are scaled with attention coefficients (α). Spatial regions are weighed by analyzing both the input features ($\mathbf{f}^s$) and the gating signal ($\mathbf{g}^{s+1}$) which is collected from a coarser scale. Grid resampling of attention coefficients is performed by trilinear interpolation.

sigmoid operation: $\sigma_2(x) = \frac{1}{1+\exp(-x)}$ (i.e., normalization function). The three linear transformations and two bias terms $\mathbf{W}_f \in \mathbb{R}^{c_s\times c_{ag}}$, $\mathbf{W}_g \in \mathbb{R}^{c_g\times c_{ag}}$, $\psi \in \mathbb{R}^{c_{ag}\times 1}$, $\mathbf{b}_\psi \in \mathbb{R}$, and $\mathbf{b}_{fg} \in \mathbb{R}^{c_{ag}}$ were applied to the gating function (2.27). $\mathbf{g^{s+1}}$ indicates the gating signal which is the final feature layer at scale $s+1$. That is, the gating signal $\mathbf{g^{s+1}}$ can be defined by the higher level feature map in U-Net, which is propagated from the lower dimensions, $\mathbf{f}^{s+1}$. c_{ag} and c_g denotes the number of channels of the AG module and of the gating signal. The linear transformations were computed using channel-wise $1\times1\times1$ convolutions. The parameters of AG modules are trained with the standard back-propagation updates [52]:

$$\frac{\partial \hat{\mathbf{f}_\mathbf{i}}^{l}}{\partial \Phi^{l-1}} = \frac{\partial\big(\alpha_i^l F_{l-1}(\mathbf{f_i}^{l-1};\Phi^{l-1})\big)}{\partial \Phi^{l-1}} = \alpha_i^l \frac{\partial\big(F_{l-1}(\mathbf{f_i}^{l-1};\Phi^{l-1})\big)}{\partial(\Phi^{l-1})} + \frac{\partial(\alpha_i^l)}{\partial(\Phi^{l-1})}\mathbf{f_i}^{l}, \quad (2.28)$$

where Φ^l denotes the set of parameters stacked to the l^{th} layer. The first gradient term is scaled with α_i^l which is an attention coefficient map.

The proposed grid attention mechanism by AG module is applied to all the skip connections that incorporate multi-scaled features. Thus, the applied attention mechanisms (i.e., AG modules) ensure multi-scaled attention which implies an ability to influence the responses to a large-to-small range of image foreground context. The overall objective function is computed by the dice loss [60]:

$$L_{agunet} = D(\tilde{\mathbf{y}}_{0.5}, \mathbf{y}) + \alpha ||W||_2^2, \tag{2.29}$$

where $\tilde{\mathbf{y}}_{0.5}$ is the binary output prediction thresholded by 0.5 after softmax, $\mathbf{y}$ indicates the ground-truth label image, W denotes the set of parameters of the network, and α is a weighting coefficient. Note that the objective function is the same as that of the V-Net [60].

The proposed attention doesn't require additional localization model in multi-staged neural networks [65, 66]. The attention gates progressively suppress feature responses in irrelevant background regions without the requirement to crop object-specific interest regions. The multi-scaled AG modules allow the model parameters in shallower layers to be updated mostly based on spatial regions that are relevant to a given task [52]. Multi-scaled feature maps were merged through skip connections to combine coarse- and fine-level dense predictions (Fig. 2.13). The progressive architecture of combining multi-scaled features was incorporated into the standard U-Net architecture to highlight salient features that are passed through the skip connections.

Auto-context neural networks

The auto-context algorithm fuses implicit shape information and low-level appearance feature to perform image segmentation [11]. Posterior distribution of the given segmentation problem is learned with the marginal distribution (i.e., classified probability map), which is further combined to learn the final classifiers. The posterior marginal is trained through image patches by calculating

Table 2.1

Employed methods in CNN-based Medical Image Segmentation Networks.

Methods	**3D U-Net** [58]	**V-Net** [60]	**DSN** [26]	**VoxRes-Net** [62]	**DenseV-Net** [63]	**AGU-Net** [52]
Residual connection	X	O	X	O	X	X
Dense connection	X	X	X	X	O	X
Deep supervision	X	X	O	O	X	O
Auto-context	X	X	X	O	X	X
Attention	X	X	X	X	X	O
Shape-prior	X	X	X	X	O	X
Post processing	X	X	O	X	X	X

the following distribution [11]:

$$p(y_i|\mathbf{x}) = \int p(y_i, \mathbf{y}_{-i}|\mathbf{x}) d\mathbf{y}_{-i}, \tag{2.30}$$

where $\mathbf{x}$, $\mathbf{y}$ present a given image and ground-truth label vector, respectively, and $\mathbf{y}_{-i}$ is a marginal set, $\{\mathbf{y} - y_i\}$. Patch representation is omitted for simplicity. Traditional feature extractors (e.g., Haar [70], HOG [71]) and classifiers (e.g., probabilistic boosting tree [72]) were used for patch-wise prediction to calculate (2.30). The algorithm iteratively solves the posterior probability with the previous marginal distribution:

$$p^{(t)}(y_i|\mathbf{x}, \tilde{\mathbf{p}}^{(t-1)}) \longrightarrow p(y_i|\mathbf{x}), \tag{2.31}$$

where $\tilde{\mathbf{p}}^{(t-1)}$ is a posterior marginal for each pixel i learned by (2.30). It is proved by the authors that the algorithm asymptotically converges to $p(y_i|\mathbf{x})$ with a discrete, iterative process. In contrast to the original work [11], the term *"context"* is used in this dissertation as a feature used in the second classifier (i.e., not shape information).

The literature of an auto-context algorithm applied to the plain neural networks is relatively low [62, 73]. The typical method is to apply a two-step inference of neural networks with identical structure [62, 73]. The first output posterior is used as a prior for the next iteration with the second neural network [62]. The iterative application of the auto-context algorithm is tedious because the neural network has to be trained by the two separate procedures. Different from the previous methods [62, 73], this work proposes a single-step training of auto-context neural network with a deeply supervising scheme [22].

2.3 Liver and vessel segmentation

In this section, a literature review of algorithms for liver and vessel segmentation is presented. First, classical methods for liver and vessel segmentation are described. The literature includes maximum intensity-based approaches [74–76] that are similar to the proposed vessel segmentation algorithm. Then, an active contour model is thoroughly reviewed which is the backbone of the proposed algorithm in this dissertation. Finally, a vessel topology-based active contour model is illustrated.

2.3.1 Classical methods for liver segmentation

The two main categories of classical image segmentation can be viewed as intensity-based and shape-based approaches. Intensity-based approaches try to delineate the object's internal distributions or contours. A graph-based optimization metric also can be regarded as an intensity-based approach. On the other hand, shape-based approaches are based on the registration method. That is, prior shapes (i.e., model database) are matched to the input to delineate the accurate contour of an input object. Note that many shape-based methods uti-

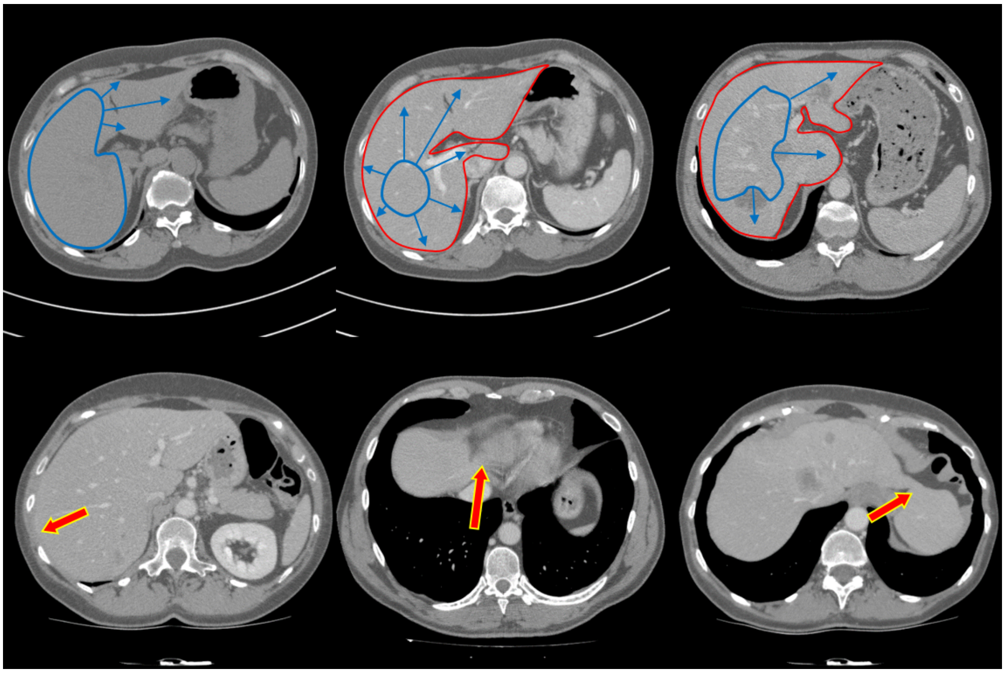


Figure 2.15: Dynamic intensity distribution and unclear boundaries of a liver. It is difficult to delineate accurate boundaries via intensity-based contour propagation.

lize intensity-based methods inside their algorithms. The major limitation of classical methods is that algorithms are parameter-dependent which implies over-fitting. Detailed strengths and weaknesses of the classics are described in the following subsections.

Local intensity-based approaches

A huge body of literature for liver segmentation has been proposed in decades [77–82]. In the beginning, local intensity-based region and contour analyses were dominant approaches [77,78]. Further, a high-level representations such as graphs [81–83] and geometric contours [79,80,84] have been proposed. Several

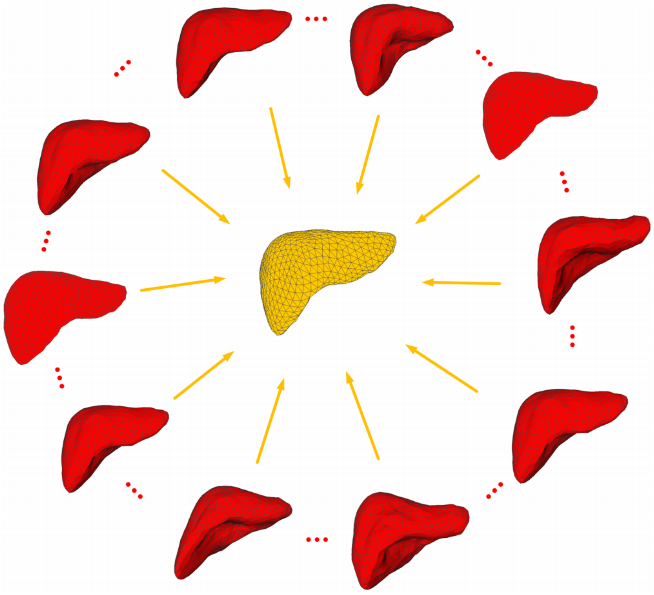


Figure 2.16: Active/statistical shape model approach. Prior shape models are fitted to the desired shape regarding the input images [89].

methods have been combined with the existing models to segment the liver more accurately [85]. In recent years, a few studies have been proposed to fuse the modern neural networks and the classics (e.g., level-set, super-pixel-based graph optimization, or region growing) [86,87]. However, even though it is empowered by neural networks, local intensity-based analysis has a basic limitation on liver segmentation due to the unclear boundaries and dynamic intensity distribution (Fig. 2.15).

Global shape-based approaches

The most successful approaches among classics were model-based (i.e., shape-based) registration methods [88]. Unlike local intensity analysis (e.g., region growing or active contour models), model-based methods employ prior shapes that can represent a liver by combining or deforming them (Fig. 2.16) [89]. Such shape-prior-based methods were more successful than the intensity-based approaches owing to shape constraints that complement the local ambiguities. However, the limitation arises from the deficiency of the training database. It has an intractable complexity to train all the inter-patient shape variations in real-world. Furthermore, even with a sufficient shape database, the model-based approach still has to overcome the accurate registration task for the final delineation of a liver. Local intensity analysis is an inevitable choice again for accurate registration. Thus, the results are highly dependent on a balanced energy functional between the local intensity-based energy and the model (i.e., shape) constraint, indicating that it is easily over-fitted to the training database. It is practically hard to set optimal balance to provide maximum performance of generalization.

2.3.2 Vascular image segmentation

In the case of vessel segmentation task, deep learning is very hard to be employed due to the lack of annotated data. Unlike other organs, vessel structures are far more tedious to manually annotate from 3-dimensional images. In decades, many classical approaches have been proposed for the task of vessel structure segmentation. The most simple and intuitive approach to perform vessel segmentation are thresholding [74, 90, 91], morphological operations [92–95], and region-growing method [96–101]. The main drawbacks of thresholding-based methods are that the optimum threshold value is very hard to determine and geometric information is not considered. Morphological operators that are similar to a matched filter approach, apply structural elements for extracting vessel topology. It is useful with a properly designed structural element. However, it is hard to fit a proper structural element to complex vascular structures with multi-scale analysis. The classical methods are computationally efficient compared to other advanced techniques, but hard to set optimal parameters. Furthermore, the algorithms typically require manual input points which make the method not fully automated. Many researchers used local geometric features of a blood vessel (e.g., tubular) to enhance vessel regions to ease segmentation problems [102–107]. Local phase-based filter [105] is also an important structural measurement capturing local features. Another common approach to enhance vessel region is using a set of spatial kernels specially designed to match vessel topology [107, 108]. Designing multiple filters to detect the vessels with different orientation and scale is a crucial part. With vessel enhancement filtering, the vessel area becomes highly contrast-enhanced as opposed to the background region. The main limitations of vessel enhancement filters are that they are hard to set scales and easily affected by noise or complex structures like in the liver vascular system.

Region-growing approach

In the region-growing method (Fig. 2.17), single or multiple seed points are selected as a starting point and iteratively expands region with certain criteria. Growing criteria have many variants such as intensity similarity, spatial proximity, and specific geometric information. The region-growing method is a computationally efficient algorithm compared to other techniques but easily suffers from noisy output or over-segmentation. More advanced region-growing methods limit growing with lower risks of leakage [98], dual object region-growing [101], and wave propagation [96, 99]. The main disadvantage of a region-growing technique is that inputs of the algorithm are typically manually processed. It is also hard to handle growing criteria with complex topological changes. There are several works to overcome such limitations by combining

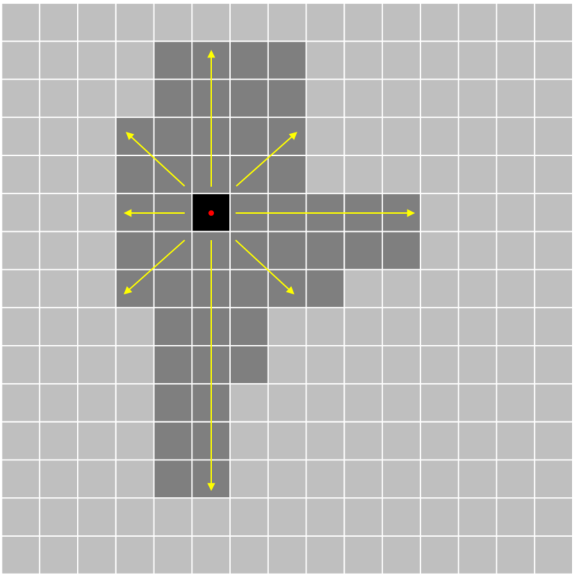


Figure 2.17: Schematic illustration of region growing method for image segmentation. A seed point (red dot) is selected as a starting point and the region iteratively grows based on certain criteria.

region-growing, morphological operation, and thresholding techniques [101].

A region-growing method is designed under the principle of connectivity. A criteria of connectivity is defined by the similarity of the intensity distribution. The main limitation arises from the fact that vessel contrast-enhancement is not guaranteed in an abdominal CT even for the portal phase image. Furthermore, manual seed points are not easy to be obtained automatically which makes the algorithm not fully automated.

Tubular structure enhancement

The enhancement of vascular structure has been the most popular approach for the task of segmentation or visualization [102, 105, 109, 110]. Among these, characterizing the local image geometry by second-order derivative information (i.e., principal curvatures of image intensities) is simple and computationally efficient. Multi-scale Hessian-based filters for vessel enhancement [102, 109] is the most common way to capture vessel geometric information by second-order derivatives. These filters use eigenvalues of a Hessian matrix to measure blob, contrast, and tubular structures. Local phase-based filters [105] are also an important structural measurement capturing local features.

As aforementioned, vessel enhancement filter is widely used for the applications of segmentation or visualization of vessels [102, 105, 106]. This approach conceives vessel enhancement as a filtering process that searches for geometrical structures that can be regarded as tubular structures [102]. Eigenvalue analysis with second-order information (i.e., Hessian matrix) for each voxel (or pixel in 2D) is used to extract structural information such as tubular. In 2D, blobness measure accounting for the eccentricity of the second-order ellipse is defined as $\mathcal{R}_B = \lambda_1/\lambda_2$ where λ_1 and λ_2 are eigenvalues of the Hessian ($|\lambda_1| \leq |\lambda_2|$). For contrast measure compared to the background, the Frobenius matrix norm is used [102]:

$$\mathcal{S} = ||\mathcal{H}||_F = \sqrt{\sum_{j \leq D} {\lambda_j}^2}, \tag{2.32}$$

where D is a dimension of the image. The vesselness function is defined using $\mathcal{R}_B$ and S measures [102]:

$$V(I) = \begin{cases} 0, & \text{if } \lambda_2 > 0, \\ \exp(-\frac{{\mathcal{R}_B}^2}{2\beta^2})(1 - \exp(-\frac{\mathcal{S}^2}{2c^2})), & \text{otherwise}, \end{cases} \tag{2.33}$$

where $I : (x, y) \longrightarrow \mathbb{R}$ is 2D image. Equation (2.33) is for enhancing bright curvilinear structures like vessel. For vesselness computation of multiple scales, multiple Gaussian filters at multiple scales are applied:

$$V(I) = \max_{s_{min} \leq S \leq S_{max}} \{V(s, I)\} = \max_{s_{min} \leq S \leq S_{max}} \{V(G(s) * I)\} \tag{2.34}$$

where s_{min} and s_{max} are the minimum and maximum scales at which relevant structures are expected to be found [102]. The degree of the scale (i.e., expected approximation of vessel width) is controlled by s, which is a standard deviation of D-dimensional Gaussian kernel applied to the image:

$$G(\mathbf{x}, s) = \frac{1}{\sqrt{2\pi s^2}^D} \exp(\frac{||\mathbf{x}||^2}{2s^2}), \tag{2.35}$$

where $\mathbf{x} \in \mathbb{R}^D$. The main limitations of vessel enhancement filters are that it is very difficult to set scales and filters are easily affected by noise or complex structures like in the liver vascular system.

Tree-based approaches

Along with tubular structure enhancement techniques, tree-like structure with a hierarchical property also inspired many researchers. This approach includes tracking [111–117], ridge traversal, and skeletonization [118,119]. Tracking methods are often fused with tubular structure measurements [112], matched filters [113], and morphological reconstruction [114]. Ridge-based methods make images as 1D elevation maps where intensity ridges approximate the skeleton of the tubular objects [118, 119]. Since ridge-based approaches detect the skeleton in tubular objects, they can be thought of as a specialized skeleton-based approach [120].

Maximum intensity-based approaches

Several maximum intensity projection (MIP) based segmentation techniques have been proposed [74–76]. The depth-buffer segmentation algorithm using MIP images was presented by [76] using Z-buffer. The authors first project 3D volume to a 2D image by MIP then they generated Z-buffer (i.e., depth buffer for each pixels corresponding to 3D position). The minimum local roughness at each point in the Z-buffer is measured by connecting neighboring points. The final step is 3D vessel construction via connecting voxels with a region-growing scheme with thresholding. Similarly, the Z-buffer segmentation (ZBS) algorithm was proposed in [75]. In this approach, MIP image segmentation is performed to generate a list of 3D seed points for further region-growing based segmentation. The authors perform Z-buffer segmentation via smooth structure detection by four principal directions of linear fits followed by thresholding. In [74], threshold-based segmentation with ZBS is proposed. The authors perform segmentation of iterative MIP images for a single direction to achieve progressive segmentation. The algorithm excludes voxels that have once projected to the MIP image in previous iterations. Recorded 3D voxel positions that correspond to MIP pixels are used to track the original voxel position of the segmented vessel. After consistency checks on the Z-buffer, a set of seed points are extracted which are used to obtain the final segmentation via volume growing.

Although making use of MIP images is an effective way to perform segmentation due to the reduction of noise variation, there are two critical drawbacks in previous methods using depth buffers [75, 76]. Firstly, the depth buffer (i.e., Z-buffer) image introduces new noise by a large variety of background depths. Secondly, the vessel region enlarges in the Z-buffer image due to the proximity of maximum intensity positions in vessel boundaries due to partial volume artifacts [121]. Noise in the Z-buffer and dilated region makes the segmentation problem very challenging. The region-growing method for the final 3D segmentation is also vulnerable because of the high noise variance in the original image. Unlike using Z-buffers, MIP images are directly used for the segmentation of vessels in [74]. The authors used a finite mixture model and the expectation-maximization algorithm to automatically obtain a global optimum threshold via maximum likelihood estimation. They iteratively project MIP images, segment vessel in MIP via global thresholding, and record corresponding vessel voxels in 3D while ignoring previously contributed voxels. This approach has an interesting feature that the 3D vessel segmentation result is solely dependent on 2D MIP image segmentation. That is, the accuracy of the MIP image segmentation plays a key role in the accurate result. However, successive iteration leads to insufficient representation of vessels in the MIP image due to the significant decrease in the number of high-intensity pixels (i.e., the number of vascular pixels) [74]. This makes the 2D segmentation problem very difficult regarding the estimation of threshold value. Furthermore, it requires an additional stopping criterion. The major drawbacks of this approach are that global thresholding leads to noisy result even with MIP image, vessel geometric information is ignored, no fine-tuning with the original 3D image are provided, and the proposed stopping criterion is solely dependent to the intensity distribution.

2.3.3 Active contour models

Active contour model (ACM) approach was first introduced in [122], and extensive research has led to the successful development of advanced variations [123–128]. The basic idea of the active contour model is to propagate a curve towards the boundaries of objects.

$$F(C) = \int_0^1 E_{internal}(C(q)) + E_{image}(C(q)) + E_{constraint}(C(q))dq, \tag{2.36}$$

where C is an explicitly parameterized curve:

$$C(q) : [0,1] \longrightarrow \mathbb{R}^2. \tag{2.37}$$

Curve iteratively evolves with energy typically designed by a combination of internal forces and external forces as in (2.36). Internal force is from a curve

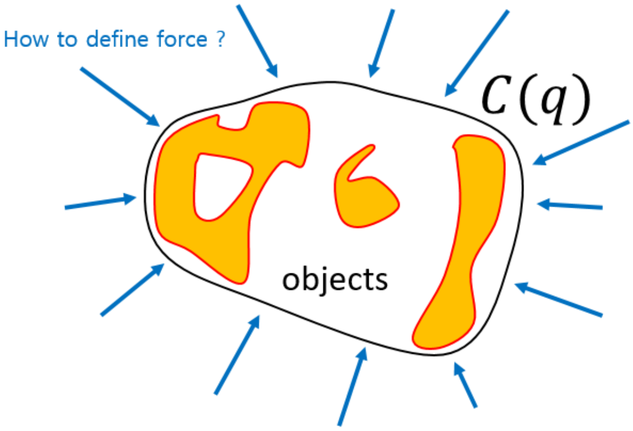


Figure 2.18: Schematic illustration of an active contour model segmentation. A curve (C) is propagated toward the boundaries of a target object. The main challenge of employing an active contour model is modeling an optimal energy functional for contour propagation.

itself that determines the geometry (e.g., smoothness of a curve) and the external force is from the image context. The energy of an image context (i.e., the external force of active contour) varies from applications to applications. The main process of the active contour approach is to minimize the energy that has a smaller value when the curve is close to target object boundaries (Fig. 2.18). Iterative propagation of a curve encloses target object boundaries by mathematical convergence. Derivation of a partial differential equation of an energy functional is different with respect to curve representations. Explicit representation of a curve (i.e., parametric active contours [122]) is computationally efficient but hard to handle topological changes during propagation and to parameterize surface in the 3D domain. Thus, an implicit representation of a curve with a level set method [129] is the most popular approach. In the level set method, the contour is implicitly represented by zero level of a higher dimensional (i.e., level set) function. Unlike parametric representation of a curve, implicit representation is more flexible to topological changes of an evolving curve because it is easy to represent multiple curves and unnecessary to explicitly parameterize a curve. A simple sign operator of level set function can determine the area of an object.

As mentioned above, the basic energy model of an active contour is composed of internal and external forces that are energy from curve geometry and image context respectively. Geodesic ACM [130] uses edge information to stop curve evolution and Chan and Vese [131] introduced new ACM without edges that uses regional information rather than edges. More improvements of level set evolution itself were made without re-initialization of level set [124] and distance regularized level set evolution [125]. Moreover, the local binary fitting (LBF) energy model [127] was presented to overcome an intensity heterogeneity of an object. In [132], the authors improved the LBF model by adopting a scale-space theory and penalizing energy functional for efficiency. To handle heterogeneous intensity distribution more robustly, analyzing local region via intensity domain transformation and multiple Gaussian distributions was presented [133]. For application to vascular segmentation, ACM made great progress with active researches [134–138].

The basic two classical models that play a key role in many variants of active contour energy models will be firstly reviewed. Then, several vessel-optimized ACMs will be reviewed in the following subsection.

Edge-based active contour model

An active contour model (ACM) is a method to propagate curves to detect boundaries of an object [122]. Either explicit or implicit representation of a curve is available. Solving ACM with implicit representation is based on the level set method [129, 139]. In the level set method, zero level set defines an object contour and we propagate a contour curve by iteratively solving Partial Differential Equation (PDE) of designed energy functional.

Geodesic ACM uses an edge-detector function depending on the gradient of an image [130]. This classical approach represented by an explicitly parameterized curve (2.37), is formulated by

$$F^G(C) = \alpha \int_0^1 |C'(q)|^2 dq - \lambda \int_0^1 |\nabla I(C(q))| dq, \tag{2.38}$$

where α, λ are real positive constants and $I : [0, w] \times [0, h] \longrightarrow \mathbb{R}$ is an input image. When minimizing the functional (2.38), the first term will keep the smoothness of a curve and the second term will make a curve converges at strong edges of an object. A general edge-detector function can be modeled by a positive and strictly decreasing function, g which is dependent on the gradient of the image. Using edge-detector function g, (2.38) can be formulated as

$$F^G(C) = \alpha \int_0^1 |C'(q)|^2 dq - \lambda \int_0^1 g\Big(|\nabla I(C(q))|\Big)^2 dq, \tag{2.39}$$

where g is a positive and decreasing function:

$$\lim_{x \to \infty} g(x) = 0. \tag{2.40}$$

For instance,

$$g(|(x)|) = \frac{1}{1 + |\nabla G_\sigma(x) * I(x)|^p}, p \geq 1 \tag{2.41}$$

where $G_\sigma * I$ is a smoother version of image I and is the convolution output of the image with the Gaussian with σ standard deviation [131]. The function g is positive at homogeneous regions and zero at edges.

An explicitly parameterized curve representation (2.39) can be replaced by an implicit representation of a curve via introducing zero level set, $C =$

$\{(x,y)|\phi(x,y)=0\}$ where ϕ is a level set function. Zero level set propagation of a curve C is given by solving the differential equation [139]:

$$\frac{\partial \phi}{\partial t} = |\nabla \phi| F, \tag{2.42}$$

where F is a speed function in normal direction of a curve. When mean curvature motion of a curve is used, F is defined by $F = div(\frac{\nabla \phi}{|\nabla \phi|})$. Level set formulation of the Geodesic ACM [130, 139] using mean curvature motion is

$$\frac{\partial \phi}{\partial t} = |\nabla \phi| (div(g(|\nabla I|) \frac{\nabla \phi}{|\nabla \phi|}) + \nu \cdot g(|\nabla I|)), \tag{2.43}$$

where ν is a constant and g is an edge-detector function same as in (2.39). Level set method iteratively solve (2.43) for time t where curve is defined by zero level set at time $t : \phi(t,x,y) = 0$.

Geodesic ACM is highly dependent on the characteristics of object edges. If the image gradient is not reliable (e.g., noisy image, very smooth edges, and discontinuous boundaries), contour propagation may not act well. However, if the gradient magnitude of a boundary object is well-bounded and consistent, contour propagation using the Geodesic model is relatively fast and accurate.

Region-based active contour model

In Chan-Vese (CV) model, which is named after the authors of [131], curve propagation is based on the Mumford-Shah segmentation techniques [131, 140]. In the fact that the CV model uses a regional term in their model, the CV model is often referred to as "region-based ACM". CV model does not use edge-function for the stopping criteria of a curve propagation [131]. In this way, an active contour model which can detect contours with or without gradients can be obtained [131]. The authors also proposed a level set formulation that achieves an automatic detection of interior contours. Furthermore, an initial curve of the proposed method can be anywhere in the image [131].

Defining input image I and evolving curve C same as in (2.38), CV energy functional is defined by

$$\begin{aligned} F^{CV}(c_1, c_2, C) =&\mu \cdot \text{Length}(C) + \nu \cdot \text{Area}(\text{inside}(C)) \\ &+ \lambda_1 \cdot \int_{inside(C)} |I(x,y) - c_1|^2 dxdy \\ &+ \lambda_2 \cdot \int_{outside(C)} |I(x,y) - c_2|^2 dxdy, \end{aligned} \tag{2.44}$$

where $\mu \geq 0, \nu \geq 0, \lambda_1, \lambda_2 > 0$ are fixed parameters, c_1 and c_2 are average values of I inside and outside C, respectively [131]. The objective becomes a minimization problem,

$$\inf_{c_1, c_2, C} F^{CV}(c_1, c_2, C). \tag{2.45}$$

Introducing Heaviside function,

$$H(z) = \begin{cases} 0, & \text{if } z \geq 0 \\ 1, & \text{otherwise (if } z < 0), \end{cases} \tag{2.46}$$

level set formulation of the CV model can be defined by [131]:

$$\begin{aligned} F^{CV}(c_1, c_2, \phi) =&\mu \int_I \delta(\phi(x,y))|\nabla\phi(x,y)|dxdy \\ &+ \nu \int_I H(\phi(x,y))dxdy \\ &+ \lambda_1 \int_I |I(x,y) - c_1|^2 H(\phi(x,y))dxdy \\ &+ \lambda_2 \int_I |I(x,y) - c_2|^2 (1 - H(\phi(x,y)))dxdy, \end{aligned} \tag{2.47}$$

where

$$c_1(\phi) = \frac{\int_I I(x,y)H(\phi(x,y))dxdy}{\int_I H(\phi(x,y))dxdy}, \tag{2.48}$$

$$c_2(\phi) = \frac{\int_I I(x,y)\Big(1 - H(\phi(x,y))\Big)dxdy}{\int_I \Big(1 - H(\phi(x,y))\Big)dxdy} \tag{2.49}$$

are the curves's internal and external averages of I, respectively [131].

Minimizing the first and the second terms mean minimizing the length and area of a curve, respectively. The third and the fourth term derives a curve C to get similar intensity distributions respectively at inside and outside of C as a function F^{CV} minimizes. Constants are typically set as $v = 0, \lambda_1 = 1$, and $\lambda_2 = 1$. The associate partial differential equation of a curve evolution in the level set formulation can be obtained by applying these constants:

$$\frac{\partial\phi}{\partial t} = \delta(\phi)\Big[\mu \cdot div(\frac{\nabla\phi}{|\nabla\phi|}) - \lambda_1 \cdot (I - c_1)^2 + \lambda_2 \cdot (I - c2)^2\Big], \tag{2.50}$$

where δ is the Dirac function defined as a derivative of the Heaviside function [131]:

$$\delta(z) = \frac{d}{dz}H(z). \tag{2.51}$$

One of the great advantages of using the CV model [131] is that the model can be successful in objects that have weak representations of edges (i.e., low gradient). As shown in Fig. 2.19, the model successfully propagates contour with regional terms without employing image gradients.

The main limitation of the CV model is that only smoothing energy and region-based energy are employed. That is, edges in the image context can be easily ignored by the intensity distribution of a region. If the intensity distribution of an internal object is homogeneous, the curve will successfully propagate regardless of edges. However, in large variance, propagation is fully dependent on background intensity distribution (i.e., outside the curve).

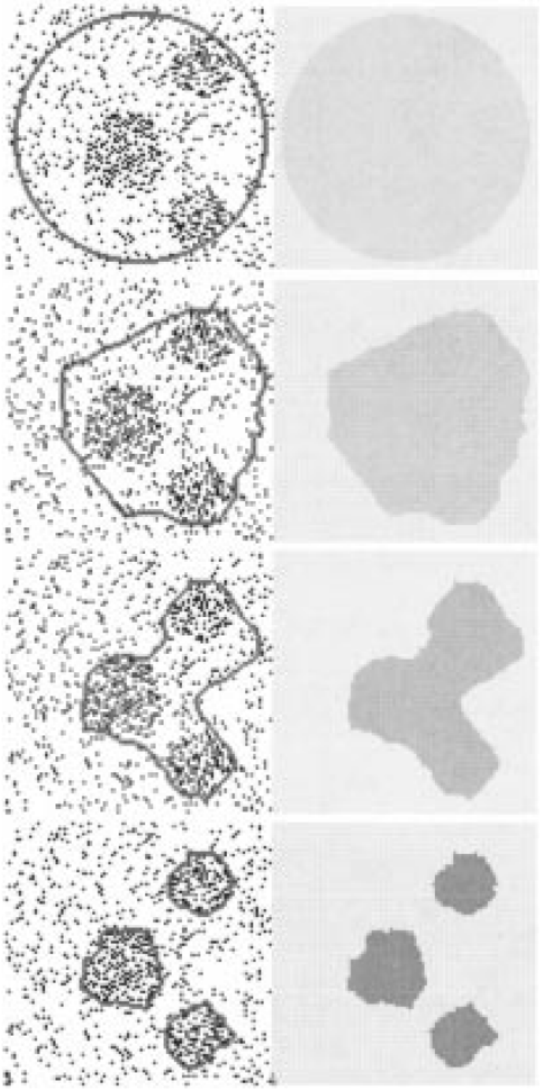


Figure 2.19: Detection of a simulated minefield with active contour without gradient [131]. The regional intensity distribution is applied for contour propagation. The model does not require the gradient of an image so that it can be applied for discontinuous images (e.g., binary).

2.3.4 Vessel topology-based active contour model

To solve the ACM in vascular segmentation problem, structural information of vessels is widely used (e.g., tubular) [134–137,141,142]: Eigen-snakes [137] that are designed for vascular segmentation with energy using directional information of vessel, center-line curve fitting method that evolves a 1D curve on a 3D domain [134], infinite perimeter model [138], vascular ACM which is using vessel enhancement filter measure and vector fields of local vessel directions for weak vessel segmentation [135], and ACM with local morphology fitting method [136]. The main drawback of an active contour approach is that initial conditions (i.e., initial contour) are hard to be set automatically. Different initial contour results in a different output that indicates many local minimum convergences. Furthermore, the noise of an image can easily affect curve evolution with image context although the energy model is aided by vessel structural analysis. Minor vessel region is also hard to find due to noise or low contrast.

Vascular active contour model

Vascular active contour model (VAC) was proposed in [135] which used a vesselness filtering method within an active contour energy functional. The authors established a single active contour model that resolves the three main tasks: 1) segmenting thick vessel region, 2) thinner and weaker vessel regions, and 3) smoothing the contour. A Gaussian mixture model is designed to robustly manage the regional term of ACM which drives a contour to include thick vessels. For the weak and peripheral branches, a vesselness filtering function (i.e., eigen analysis of the Hessian matrix) is employed in a multi-scaled manner. The vascular vector field, which is generated based on the vesselness measures, drives an external force to supplement the active contour to penetrate into the thinner and weak vessels [135]. A dual curvature strategy is added to smoothen the surface of the vessel without changing its shape [135].

A Gaussian mixture model (GMM) is employed in the region competition-based energy term [135]. Supposing there are K objects and that the statistical distribution parameters of these objects are represented by α_i with $i = 1, ..., K$ [135]. The segmentation can be defined by assigning each pixel to a corresponding object. The typical energy function is then defined as follows [131,135,143]:

$$E[\Gamma_i, \alpha_i] = \sum_{i=1,...,K} \Big\{ -\mu_I \int_{\Omega_i} \text{log} P(I(x)|\alpha_i)dx + \mu_c \int_{\Gamma_i} ds \Big\} \tag{2.52}$$

where $\{\Omega_1, ..., \Omega_K\}$ are all the regions in the image, $\Omega = \cup_{i=1}^{K} \Omega_i$, $\Omega_i \cup \Omega_j = \phi$ if $i \neq j$, where α_i represents the statistical distribution of the region i in a broad sense, where Γ_i is the union of the closest boundaries: $\Gamma_i = \partial \Omega_i$ and $x = [x_1, x_2, ...]^T$ and where μ_I and μ_c are constants preserving the balance between the region and the boundary energy [135].

Supposing that the target object consists of several objects, $\Omega_o = \cup_{i=1}^{m} \Omega_i$, then the background can be represented by $\Omega_b = \cup_{i=m+1}^{K} \Omega_i$. The segmentation problem can be finally defined by minimizing energy functions: $(I(x)|\Omega_o) = \max_{i=1,...,m} P(I(x)|\alpha_i)$ and $P(I(x)|\Omega_b) = \max_{i=m+1,...,K} P(I(x)|\alpha_i)$ [135]. By integrating two objectives, the energy function can be written as follows:

$$\begin{aligned} E[\Gamma_o, \alpha_i] = &\mu \int_{\Gamma_o} ds - \mu_I \Big\{ \int_{\Omega_o} \log \max_{i=1,...m} P(I(x)|\alpha_i)dx \\ &+ \int_{\Omega - \Omega_o} \log \max_{i=m+1,...,K} P(I(x)|\alpha_i)dx \Big\}. \end{aligned} \tag{2.53}$$

The level set formulation of (2.53) can be represented by employing the Euler-Lagrange differential equation [144]:

$$\frac{\partial \phi}{\partial t} = \delta(\phi) \Big\{ \mu_c \text{div}(\frac{\nabla \phi}{|\nabla \phi|}) + \mu_I [\log \max_{i=1,...,m} P(I|\alpha_i) - \log \max_{i=m+1,...,K} P(I|\alpha_i)] \Big\} \tag{2.54}$$

where $\delta(\phi)$ is a Dirac function [135]. Assuming that the gray values of the objects in the image satisfy a Gaussian distribution, the histogram can be rather well described by a mixture of Gaussian curves $\sum_{i=1}^{K} \beta_i G(\mu_i, \sigma_i)$ with β_i, μ_i, and σ_i are the prior probability, mean, and standard deviation of the object class, Ω_i, respectively [135]. The statistical distribution for an object, $\alpha_i = \beta_i, (\mu_i, \sigma_i)$, can be estimated for all the classes based on the expectation maximization (EM) algorithm:

$$\frac{\partial \phi}{\partial t} = \delta(\phi)\Big\{\mu_c \mathrm{div}(\frac{\nabla \phi}{|\nabla \phi|}) + \mu_I v_I(I)\Big\}, \tag{2.55}$$

where

$$v_I(I) = -\min_{i=1,\ldots,m}\left(\frac{(I-\mu_i)^2}{2{\sigma_i}^2} + \mathrm{log}\sigma_i\right) + \min_{i=m+1,\ldots,K}\left(\frac{(I-\mu_i)^2}{2{\sigma_i}^2} + \mathrm{log}\alpha_i\right). \tag{2.56}$$

The class parameters are estimated by the EM algorithm before the contour propagation.

The authors also proposed replacing $\delta(\phi)$ by $|\nabla \phi|$ that enables the zero level set to move in a broader band and hence drives faster convergence [135]. To stabilize and boost the convergence of contour on sharp edges, a speed-controlling term is proposed [135]: $1/(1 + \alpha|\nabla I \cdot \nabla \phi|)$. The speed-controlling term will have a very low value when the magnitudes of the vectors ∇I and $\nabla \phi$ are high and have similar directions [135]. The term penalizes the progression of an active contour on strong edges that achieves more robust convergence. The level set formulation of the proposed energies can be written as follows:

$$\frac{\partial \phi}{\partial t} = \frac{|\nabla(\phi)|}{1 + \alpha|\nabla G(I) \cdot \nabla \phi|}\Big\{\mu_c \mathrm{div}(\frac{\nabla \phi}{|\nabla \phi|}) + \mu_I v_I(I)\Big\}. \tag{2.57}$$

The vascular vector field is also introduced by the authors [135]. As mentioned above, the vascular vector field is based on the multi-scale vesselness filtering to drive a contour propagation into weak and thin vessels. The magnitude of the vector field $\vec{V}(x)$ is a function of the vesselness measure, $R(x)$, where both $R(x)$ and $\vec{V}(x)$ are derived from the eigen analysis of the Hessian matrix H_I of the 3D image [135].

A tubular structure can be identified by a principal component vector via eigenvalue analysis of H_I (i.e., λ_1, λ_2, and λ_3 are eigenvalues of H_I with $|\lambda_1| \leq |\lambda_2| \leq |\lambda_3|$ and v_1, v_2, and v_3 are the corresponding eigenvectors, respectively). The direction along the vessel can be represented by v_1 which indicates a minimal variation in intensity, while v_2, v_3 will form a plne perpendicular to v_1 [135]. The vector field, $\vec{V}(x)$, are calculated over all voxels x in the image:

$$\vec{V}(x) = \begin{cases} \vec{v_1}, & \text{if } R(x) > \tau, \\ 0, & \text{otherwise,} \end{cases} \tag{2.58}$$

where $R(x)$ is the vesselness measure and τ is a threshold value for the vascular structures [135]. The authors used Rashindra's vesselness measure [105]:

$$\text{R(x)=} \begin{cases} 0, & \text{if } \lambda_2 > 0 \text{ or } \lambda_3 > 0, \\ \left(1 - e^{-A^2/(2\alpha^2)}\right) \cdot e^{-\beta^2/(2\beta^2)} \cdot \left(1 - e^{-S^2/(2\gamma^2)}\right) \cdot e^{-\frac{2c^2}{|\lambda_2|\lambda_3^2}}, & \text{otherwise} \end{cases} \tag{2.59}$$

where $A = |\lambda_2|/|\lambda_3|$, $B = |\lambda_1|/\sqrt{|\lambda_2\lambda_3|}$, and $S = \sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}$. The term A identifies whether the structure is planar or tubular, B accounts for the blob structure, and S represents an overall difference between vessel and background [135].

Scale is a very sensitive parameter for vessel structure analysis via vesselness measures [102]. The thickness of a vessel (i.e., diameter) affects the final response of vesselness filters. The authors in [135] simply applied multi-scaled analysis for the vascular vector field similar to [102]. The one with the highest value is chosen and the vector at the corresponding scale is taken as the final vector in the proposed vascular vector field [135]:

$$\vec{V}(x) = \{\vec{V_\sigma}(x)|\alpha_\sigma R_\sigma(x)\} \tag{2.60}$$

where

$$R(x) = \max_{\sigma_{min} \leq \sigma \leq \sigma_{max}} \{\alpha_\sigma R_\sigma(x)\}, \tag{2.61}$$

and α_σ is the scale's weight that is related to σ [135]. The directions of a vector field are further adjusted to be aligned for the surface normal direction of active contour [135]:

$$\vec{v}(x) = \begin{cases} \vec{V}(x), & \text{if } V(x) \cdot \nabla \vec{\phi}(x) \geq 0, \\ -\vec{V}(x), & \text{otherwise.} \end{cases} \tag{2.62}$$

The modification of directions is to make an active contour expand towards the small vessels.

Finally, the magnitude of the vector field is employed based on the vesselness measure (2.61). The main purpose of introducing magnitudes is to drive an active contour to get a high speed inside the thinner vessels and to get a lower speed on the edge of the vessels [135]. By introducing an additional magnitude function as

$$f_\epsilon(R(x)) = \frac{1}{2}(1 + \frac{2}{\pi}\arctan(\frac{R(x) - \epsilon}{\epsilon})), \tag{2.63}$$

where ϵ is a threshold of the vesselness measure, the final evolution equation of the vascular vector field-driven active contour can be defined as follows [135]:

$$\frac{\partial \phi(x)}{\partial t} = f_\epsilon(R(x)) |\vec{V}(x) \cdot \nabla \phi(x)|. \tag{2.64}$$

If $R(x)$ is larger than ϵ, $f_\epsilon(R(x))$ reaches its highest value quickly; inversely, it reaches zero quickly [135].

To make a smooth surface while preserving the shapes, an additional curvature term is presented. A typical smoothness constraint based on mean curvature is $\text{div}(\nabla\phi/|\nabla\phi|) = -2H$, where H represents a mean curvature (i.e., $H = (k_1 + k_2)/2$, where k_1, k_2 are maximal, minimal principal curvatures, respectively). The H mainly depends on k_1 for thin vessels which forces the surface to shrink [135]. The decision whether to use mean or minimal principal curvature can be introduced as a new smoothing constraint [31]:

$$f_c(R(x)) = \begin{cases} -k_2, & R(x) > c, \\ -H, & \text{otherwise} \end{cases} \tag{2.65}$$

with c being a threshold of vesselness measure. In this way, the magnitude of vesselness filter responses is employed to give a criterion for whether curvature to use. When strong vesselness response is monitored (i.e., weak and thin vessel region), minimal curvature is adopted to preserve the shrink of an active contour.

By compiling all the aforementioned three methods, the final evolution equation is defined as follows [135]:

$$\frac{\partial \phi}{\partial t} = \frac{|\nabla \phi|}{1 + \alpha |\nabla G(I) \cdot \nabla \phi|}(\mu_I v_I(I) + \mu_c f_c(R)) + \mu_{\vec{V}} f_\epsilon(R) |\vec{V} \cdot \nabla \phi| \tag{2.66}$$

where μ_I, $\mu_{\vec{V}}$, and μ_c are constants to make a balance between the three terms.

VAC algorithm [135] presented a new active contour model which embedded vesselness function [102] to enhance the capability of segmenting weak and thin vessel area. The GMM method improves the energy term of foreground intensity distributions that can affect the thick vessel regions. A new curvature adaptation depending on vesselness measure showed an adaptive method to preserve thin vessels while maintaining smoothness. However, it is very hard to set many parameters to be optimal to make the VAC model successfully segment the vessel area. It is tedious to set optimal parameters, such as thresholds, for all the input images. Moreover, the GMM model relies on the training, which indicates dynamic intensity distributions of input images (e.g., multi-phases, low contrast-enhanced images) are not likely to be successful by the VAC model.

2.4 Motivation

Convolutional neural networks (CNNs) are the most successful architecture to perform medical image segmentation in recent years. However, the algorithm requires a large dataset of manually annotated images to train neural networks in a supervised manner. Manual annotation of 3D medical images is a time-consuming and difficult task. The degree of difficulty increases with complex objects with weak representations such as liver and vessels. Thus, to employ CNNs to the actual clinical systems, the performance of generalization must be improved with minimum training datasets.

The generalization problem has been a well-known task for the actual deployment of neural networks in many fields. Various studies have been conducted to obtain a high generalization performance such as weight decay, drop out [3], transfer learning [4], data augmentation [5], domain adaptation [6, 7], and regularization of loss functions [8]. However, none of those techniques present a domain-specific knowledge, i.e., curriculum learning, to improve the generalization performance. Furthermore, few works present a study of generalization on image segmentation tasks. To overcome the difficulty of improving generalization, a domain-specific technique is highly required. Therefore, an architectural way to improve the generalization is suggested in this dissertation for liver segmentation task (chapter 3).

As mentioned above, manual annotation of complex liver vessels is hard to be obtained. Thus, a human-designed algorithm is required to perform accurate liver vessel segmentation. In contrast-enhanced abdominal CT images, vessels are enhanced by a contrast media mostly in the portal phase (among the four phases). However, the degree of enhancement differs from patient to patient, and noise on CT images hinders the accurate segmentation of vessels. Based on the facts, previous approaches assuming regional connectivity or structures (e.g., trees) are difficult to be applied in every cases. Furthermore, thin vessels are hard to be identified in the original image due to its low contrasts.

Although a vessel enhancement filtering is the most promising method for vessel segmentation task, it is difficult to set global scale parameters of a filter for both thick and thin vessels. A complex 3D vessel structure also degrades accurate filtering responses. In that sense, although an active contour model performs well on vessel segmentation, designing an energy functional based on vessel enhancement filter of global scales is prone to be unsuccessful. Thus, to segment all the minor vessels successfully, it is important to represent minor details while preserving the thick vessels. A novel dense vessel candidate points are introduced for the desired task in this dissertation (chapter 4).

Chapter 3

Liver Segmentation via Auto-Context Neural Network with Self-Supervised Contour Attention

3.1 Overview

Segmentation of a liver is an essential prerequisite for various clinical applications such as volumetric measurement of liver and its structural analysis. Furthermore, accurate liver segmentation allows later algorithms such as vessel segmentation to be easier and more automated. This is because the foreground area of liver acts as a solid prior to remove unnecessary background areas. From various perspectives, such as engineering and clinical needs, automated segmentation of a liver is the most important and essential part of computer-aided diagnosis systems.

In this chapter, a neural network architecture that exploits an auto-context algorithm [11] and the sparse contour attention metric is presented. An auto-

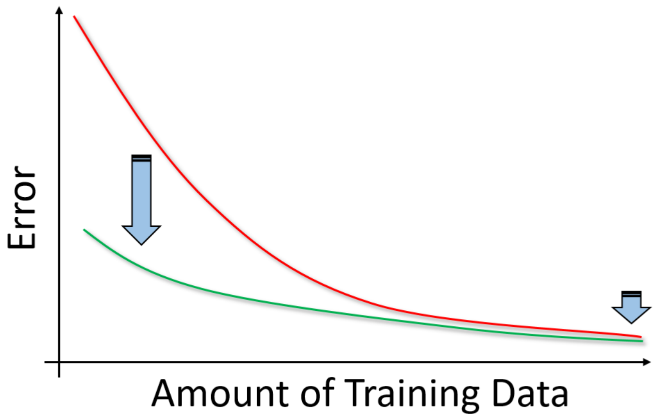


Figure 3.1: The main objective for deep learning-based liver segmentation. As the total amount of training data decreases, the overall error increase in nature (red and green lines). It is difficult to suggest a certain criterion on the number of training data regarding desired test accuracy. It is important to provide a good performance of generalization whether sufficient or insufficient training data is provided. The proposed algorithm presents a method to reduce test errors even when the small number of training data is provided.

context algorithm is implemented by a deeply supervising fashion to embed liver posterior analysis to the network which acts as a prior in the following layers. In such a way, an auto-context algorithm is designed in a single-passing neural network. In extension to the baseline auto-context neural network, contour attention mechanism is applied to improve the fine delineation of the target liver.

The major objective of the proposed architecture is to improve the performance of generalization and accuracy (Fig. 3.1). Analyzing and improving generalization performance is very important because one of the fatal drawbacks of deep learning is the inability to provide solid guidelines for generalization performance. Especially in medical image segmentation tasks, it is difficult to obtain large scale annotated images, so it is necessary to improve generalization performance. Thus, in this dissertation, a neural network that shows a great performance of generalization is proposed. The proposed method shows that overall accurate shape inference is the most important factor for improving generalization performance. Without inferences of overall shape, the network degrades performance by making many false predictions. Robust shape inference is obtained by introducing a shape-prior, not through previously trained shapes, but through the network's trainable parameters. The shape inference process is implemented by the auto-context algorithm on a single network. For more robust shape inference, an additional high-level residual connection is proposed which allows accurate inference without the construction of complex networks that require many parameters. Rich studies will be presented regarding the performance of generalization by comparing the proposed method with several state-of-the-art networks and self-ablations.

Another main contribution of the proposed method is the improvement of accuracy through the contour scheme. It is very difficult to plant the concept of contour in fully convolutional neural networks because of the many ambiguous boundaries (i.e., unclear, homogeneous regions) in a liver contour on abdominal CT images. Ambiguous boundaries are difficult to be trained even for neural networks that have intractable complexity. Forcing the network to accurately represent the entire contour of a liver leads to degradation of overall accuracy including the increase of false positive predictions. This shows that a simple supervising metric, such as supervision of ground-truth contour, is not feasible. In the proposed network, a self-supervised learning method is proposed instead of using the entire contour. The proposed self-supervised learning method induces the network to learn by itself, focusing on the contour of the wrongly predicted

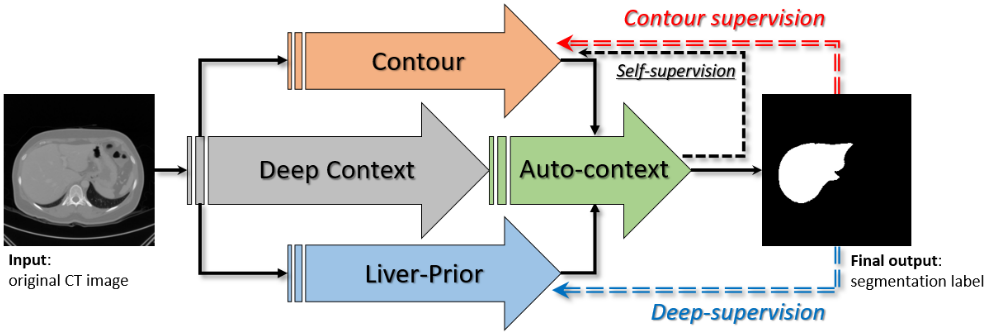


Figure 3.2: The overall architecture of the proposed neural network. The network comprises three sub-routes: 1) deep context and auto-context, 2) liver-prior, and 3) contour. The liver-prior branch is deeply supervised to predict the coarse posterior of a liver. The predicted posterior is used as a prior for the following auto-context prediction. The contour branch is also deeply supervised by a self-supervising scheme based on the prediction of the network itself.

region. The overall architecture is described in the following subsections.

3.2 Single-pass auto-context neural network

The architecture of the proposed network is composed of several sub-networks: deep context, liver-prior, auto-context, and contour. As shown in Fig. 3.2, the liver-prior branch is deeply supervised by the ground-truth liver and combined to the features of deep context. The liver-prior sub-network implies a shape-prior which formulates an auto-context algorithm of the network. A posterior of liver-prior sub-network acts as a prior to the following sub-networks. In addition to the auto-context baseline, the network is extended by a contour branch which is self-supervised based on the ground-truth contour and the final prediction of the network. The final prediction, for each training iteration, is utilized to penalize the ground-truth contour which makes sparse contour supervision. It is based on the confident penalization that builds a self-supervising mechanism for which a contour sub-network to be trained to attend the failures regarding contour regions.

In the following subsections, two non-linear convolutional modules, which are the building blocks of the networks, are firstly presented. Then, the architecture of an auto-context neural network and self-supervising contour attention mechanism will be illustrated.

3.2.1 Skip-attention module

The Skip-attention block (Fig. 3.3) is first used to extract common features (i.e., shared features in the following layers). The features are thereafter fed to the sub-networks: liver-prior, context, and contour (Fig. 3.6). Skip-attention block is composed of non-linear transformation series: depth-wise separable convolutions [23], batch normalization [18], and ReLU non-linear activation function [19] (Fig. 3.3). These transformations are skip connected for feature reuse. Depth-wise separable convolutions [23] is introduced in the Skip-attention rather than bottleneck layers [24] or compression layers [21] for more efficient use of parameters. A channel-wise attention mechanism is applied to the final output to employ channel-wise attention similar to [51]. Unlike [51], for simplicity, trainable channel-wise attention vector is directly applied which is multiplied for each channel. The formulation of the Skip-attention block is as follows:

$$S(\mathbf{x}) = \left[F_0(\mathbf{x}), \sigma_{relu}(b((F_0(\mathbf{x}) * \ddot{\theta}_1^{48}), \gamma_1))\right] \otimes_c \sigma_{softmax}(\mathbf{c}_{96}), \tag{3.1}$$

where F_0 indicates the first non-linear series defined as

$$F_0(\mathbf{x}) = \sigma_{relu}(b((\mathbf{x} * \theta_0^{48}), \gamma_0)), \tag{3.2}$$

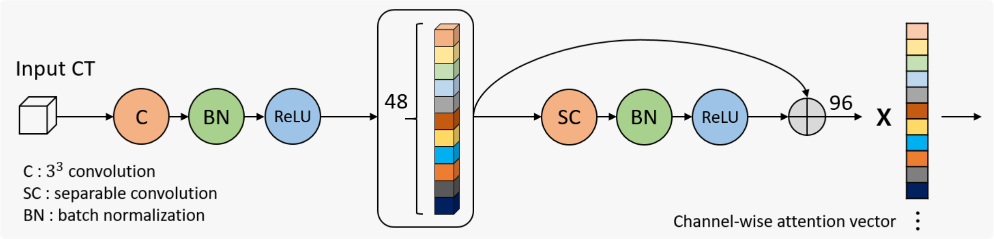


Figure 3.3: Skip-attention layer (common feature extraction module). The layer is composed of non-linear transformation series: depth-wise separable convolutions, batch normalization, and ReLU non-linear activation function. The first convolution is applied without separable convolutions due to a single-channeled input. The intermediate features are skip-connected by concatenation. A trainable channel-wise attention vector is employed for the final output of the layer. The number of output feature is 96.

where $\mathbf{x}$ is an input image and θ_l^n denotes weights of l^{th} convolutional kernel where n indicates the number of output channels. b, γ_i, σ_{relu}, and $\sigma_{softmax}$ represent batch normalization, scale parameter, rectified linear unit, and softmax operator, respectively by the same notation as specified in (2.3). $\ddot{\theta}$ denotes a separable convolution (Fig. 3.4) in (3.1). $[\mathbf{x_i}, \mathbf{x_j}]$ in (3.1) indicates a channel-wise concatenation of the features. The final output feature map of the Skip-attention block is channel-wise multiplied ($\otimes_c$) by the attention vector of dimension 96 (i.e., $\mathbf{c}_k \in \mathbb{R}^k$). That is, a relative importance among channels are weighed by the channel-wise attention vector which is trainable ($\mathbf{c}_{96}$).

The proposed Skip-attention module is designed to extract common features that are used in all the following sub-layers. A single skip connection by concatenation is employed for shallow but effective feature reuse. A channel-wise attention vector is a vector that is composed of trainable scalar parameters that are trained for channel-wise weighting. As defined in (3.1), the vector goes

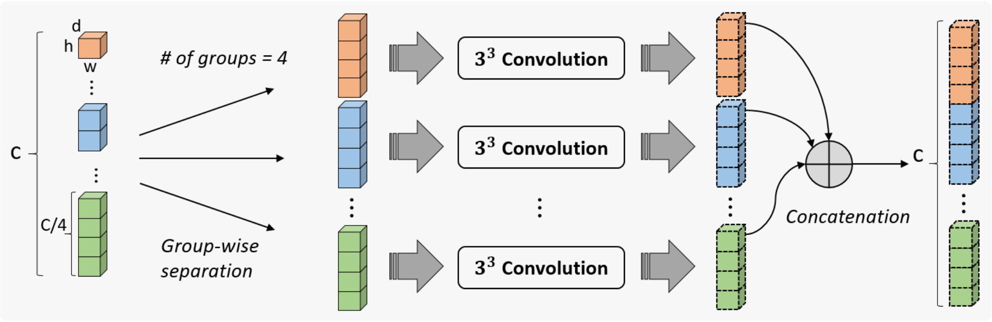


Figure 3.4: Depth-wise separable convolutions. The input channels are separated by groups and are convolved separately. The final output is a concatenation of all groups. The number of groups is 4 in the proposed network.

through the softmax operator for a channel-wise multiplication. In this way, a channel-wise attention mechanism can be applied in a more structured way. It is indeed different from individual trainable bias parameters in the convolutional operator.

Separable convolutions (Fig. 3.4), applied to the second convolutional operator in the Skip-attention module (Fig. 3.3), reduce the number of parameters within a single convolutional layer. As shown in Fig. 3.4, separable convolutions perform channel-wise separated convolutions and concatenate each of the outputs. The outcome of the separation is a reduction of parameters. For example, if the number of input channels was given by 128, a normal convolution requires 442,368 parameters for a given convolutional layer (i.e., $3^3 \times c^2$, where c is the number of channels). On the other hand, separable convolutions, with the number of groups of 4, requires 110,592 parameters that are a quarter of the normal version (i.e., $3^3 \times \frac{c}{n}^2 \times n = 3^3 \times c^2 \times \frac{1}{n}$). Note that a reduction of the number of parameters can be seen as an improvement in the performance of generalization based on a reduction in complexity (i.e., intrinsic dimension) of a network.

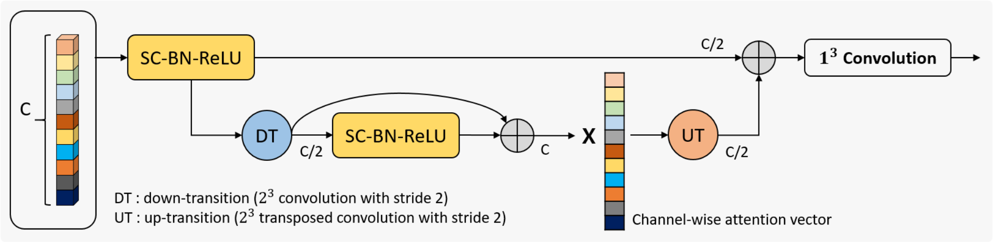

C
SC-BN-ReLU
C/2
1^3 Convolution
DT
C/2
SC-BN-ReLU
C
X
UT
C/2
DT : down-transition (2^3 convolution with stride 2)
UT : up-transition (2^3 transposed convolution with stride 2)
Channel-wise attention vector

Figure 3.5: V-transition layer. A structured non-linearity module (i.e., SC-BN-ReLU) is composed of a series of separable convolutions, batch normalization, and rectified linear unit. A multi-scaled feature analysis applied by down-transition via 2^3 convolution with stride 2 and up-transition via 2^3 transposed convolution with stride 2. A skip-connection and channel-wise attention vector are employed in the lower resolution similar to the Skip-attention block. The final output of the layer is generated by a 1^3 convolution operator applied to the concatenated features.

However, it is not always the case because it is hard to model the complexity of neural networks, and it is not linearly related. The performance of generalization and complexity of network are more dependent on task-dependent models (i.e., structures) of neural networks.

3.2.2 V-transition module

The V-transition is a small U-Net-like module designed for multi-scale analysis (Fig. 3.5). The V-transition is a building block of the network which is stacked in either a parallel or sequential manner (Fig. 3.6). A structured non-linearity module is composed of a series of separable convolutions, batch normalization, and rectified linear unit (Fig. 3.5). A multi-scaled feature fusion is employed by using down-transitioned resolution (Fig. 3.5). The down-transition operator halves the resolution of the input via 2^3 convolution with stride 2. A relative receptive field doubles in the lower resolution which derives macroscopic feature extractions. Layers in the lower resolution are designed similar to the Skip-attention module that is composed of a skip connection and channel-wise attention. Conversely, an up-transition restores the dimensions via a de-convolution (i.e., transposed convolution). By contracting and expanding paths, the V-transition layer can extract more multi-scaled features (i.e., higher receptive field). The final output of the V-transition layer is generated by a 1^3 convolution operator applied to the concatenated features of multi-scaled routes. The number of output channels is specified in Fig. 3.6.

3.2.3 Liver-prior inference and auto-context

As introduced in the previous sections, the proposed network architecture is composed of three main branches: residual shape-prior, context, and contour, i.e., the blue, gray, and orange dotted boxes in Fig. 3.6, respectively. The shape-prior network is deeply supervised to inference the ground-truth liver. The trained posterior (i.e., the output of the shape-prior network) is used as a prior for the remaining network. Deep features that are trained by the context network are concatenated to the prior for the final auto-context fusion. Besides, the contour attention branch is also deeply self-supervised with a ground-truth contour regarding the output of the network. The contour features are trained by a penalized classification scheme and further enhanced the network to better delineate liver boundaries. The two different non-linear modules are used in the proposed network (i.e., the Skip-attention block (Fig. 3.3) and V-transition layer (Fig. 3.5)).

The base architecture of the proposed network is an auto-context algorithm. Instead of stacking very deep neural layers, the proposed network uses multiple shallow stacks of layers (Fig. 3.6). The liver-prior and context layers are

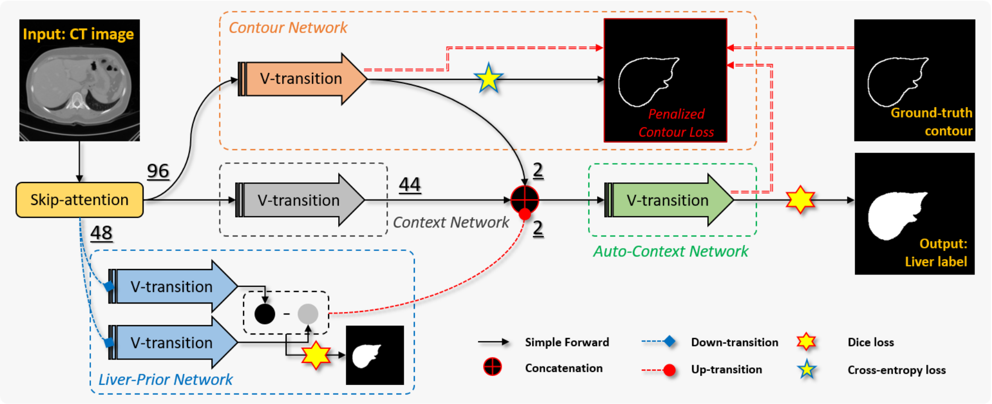
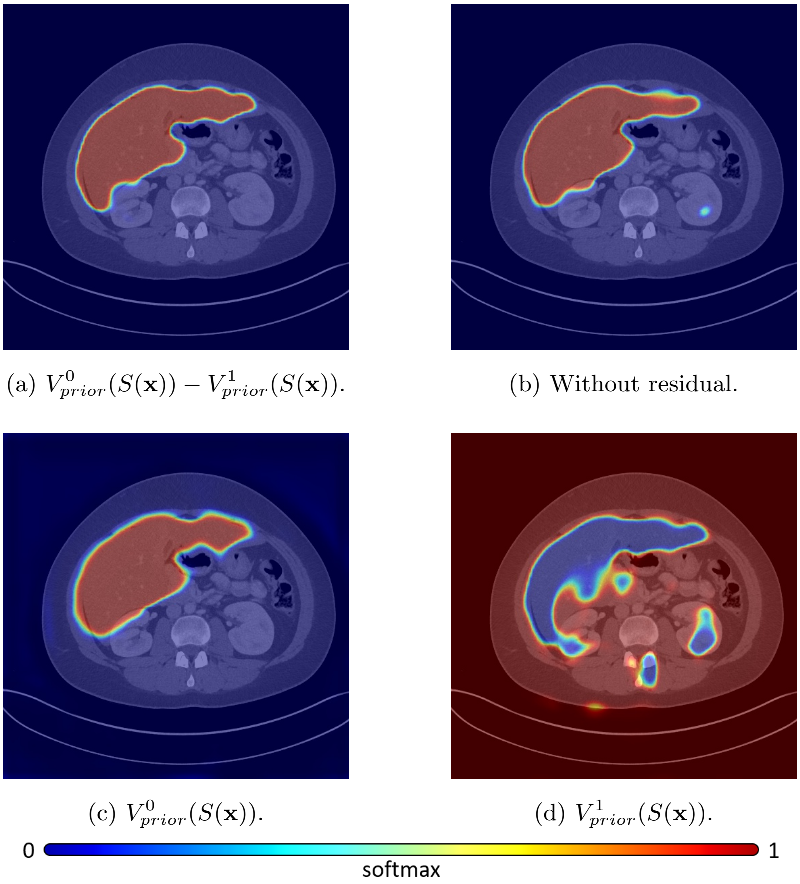


Figure 3.6: The architecture of the proposed 3D volumetric fully convolutional network. Stacked V-transitions form a base architecture with multiple skip connections. The red (i.e., circled arrows) and blue arrows (i.e., squared arrows) indicate up- and down-transition layers, respectively. The red and blue dotted boxes represent the contour and prior transitions, respectively. The two transitions are deeply supervised by the contour and ground-truth images. The final output prediction is achieved by combining all the features. All the images are displayed as 2D for simplicity.

composed of V-transition layers that are a small U-Net-like [59] module that includes down and up transitions together with skip connection (Fig. 3.5). The detailed architecture of the V-transition is visualized in Fig. 3.5. The overall architecture also uses down-sampled resolution in liver-prior sub-network. It is designed to extract more macroscopic features for overall shape estimation. Including down-transitions in the V-transition layer, the original input image is down-sampled by the factor of 4 in total. The estimated liver-prior is further up-sampled for further concatenation.

The two identical shape transitions are used in the liver-prior part to subtract each output prediction at a higher level (blue dotted box in Fig. 3.6).



(a) $V^{0}_{prior}(S(\mathbf{x})) - V^{1}_{prior}(S(\mathbf{x}))$.

(b) Without residual.

(c) $V^{0}_{prior}(S(\mathbf{x}))$.

(d) $V^{1}_{prior}(S(\mathbf{x}))$.

Figure 3.7: Example liver prior estimations by the residual inference. (a) The final inference using dual inferences (i.e., residual) (c) and (d). (b) Inference without residual. The output of (b) is obtained by sequentially stacking the two prior estimations (i.e., V-transitions) by preserving the number of intermediate features.

The output is deeply supervised with the ground-truth label image. The dual-passing architecture effectively learns mutually complementary features for the accurate inference of the liver posterior (an example visualization is presented in Fig. 3.7). The objective function for deep supervision is as follows:

$$L_{prior} = D\big((V^0_{prior}(S(\mathbf{x})) - V^1_{prior}(S(\mathbf{x}))), \mathbf{y_{dl}}\big), \tag{3.3}$$

where D indicates the soft dice loss [60], V^i_{prior} denotes i^{th} V-transition in liver-prior sub-network, and $\mathbf{y_{dl}}$ represents the ground-truth liver label image at down-scaled resolution. Finally, the output feature map is concatenated to the context features (i.e., output of the context transition; $V_{context}(S(\mathbf{x}))$) and passes through an auto-context transition (V_{auto}) for final refinement. The objective function of the final output layer can be defined as follows:

$$\mathrm{L}_{final} = D\Bigg(V_{auto}\Big(\Big[V_{context}(S(\mathbf{x})), \big(V^0_{prior}(S(\mathbf{x})) - V^1_{prior}(S(\mathbf{x}))\big), V_{contour}(S(\mathbf{x}))\Big]\Big), \mathbf{y_l}\Bigg), \tag{3.4}$$

where $V_{contour}$ indicates the contour V-transition described in the following subsection and $\mathbf{y_l}$ denotes the ground-truth liver label image. The proposed network architecture with multiple branches aids the context features to learn effective complements that can aid the final auto-context transition.

The proposed single-passing auto-context neural network exploiting high-level residual connection suggests that deepening or widening the neural network is not the only answer for complex tasks. Stacking layers sequentially makes it difficult to use parameters effectively, and especially discriminatively, and further degrades the performance of generalization of the network. After extracting common and reusable features by Skip-attention (Fig. 3.3) for multiple network branches, each branch learns complementary or different features for the high-level tasks. The further experimental section will study the ablation of high-level residual connection that describes the effectiveness of the architecture compared to a sequential architecture.

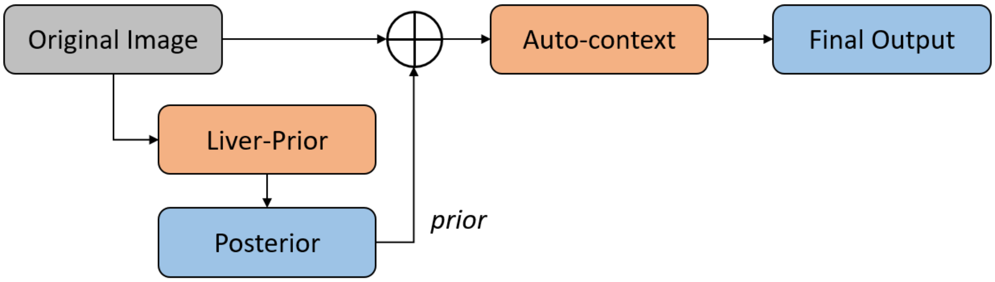


Figure 3.8: Schematic workflow of the auto-context framework of the proposed network. The posterior of liver is embedded in the network and acts as a prior in the following layers.

3.2.4 Understanding the network

Let vectors $\mathbf{x} = \{x_i \in R, i \in \mathbb{R}^3\}$ and $\mathbf{y} = \{y_i \in \{0,1\}, i \in \mathbb{R}^3\}$ represent the input image and ground-truth label, respectively. The objective of the given segmentation problem is to determine the optimal solution for modeling a conditional probability distribution, $p(\mathbf{y}|\mathbf{x})$, by maximizing a posterior (MAP):

$$\theta^* = \arg\max_{\theta} p(\mathbf{y}|\mathbf{x};\theta) = \arg\max_{\theta} p(\mathbf{x}|\mathbf{y};\theta)p(\mathbf{y}). \tag{3.5}$$

where θ is a parameter set for classifiers. However, it is very difficult to model the likelihood (i.e., $p(\mathbf{x}|\mathbf{y};\theta)$) and prior (i.e., $p(\mathbf{y})$), and solving the decomposed posterior with a generative approach easily yields inaccurate results mainly owing to the difficulty of likelihood and prior estimation. The proposed network iteratively solves the posterior directly with the auto-context method [11]. In auto-context, the previous classification map is used as a shape feature (i.e., term “context” in the original paper) for additional classification. Setting t as a discrete time value, auto-context is formulated as

$$p^{(t)}\Big(\mathbf{y}|\mathbf{x}, p^{(t-1)}(\mathbf{y}|\mathbf{x};\theta_{(t-1)});\theta_t\Big) \longrightarrow p(\mathbf{y}|\mathbf{x};\theta^*). \tag{3.6}$$

Unlike in the previous approaches [11,73], shape feature extraction procedure is embedded within a single-passing neural network (Fig. 3.8). The output of the proposed network for time t can be formulated as

$$p^{(t)}\Big(\mathbf{y}|\mathbf{x}, \tilde{p}^{(t)}(\mathbf{y}|\mathbf{x};\theta_t);\theta_t\Big), \tag{3.7}$$

where $\tilde{p}$ is a probability map of shape-residual sub-network (i.e., the blue dotted box in Fig. 3.6). Applying deep supervision (i.e., auxiliary classifiers), a single-passing neural network could be obtained by embedding a previous posterior. Thus, the architecture avoided using separated classifiers and storing previous classification maps.

3.3 Self-supervising contour attention

Edge is unquestionably the most important feature for the accurate object segmentation. From the perspective of contour delineation, the task of object segmentation in images can be achieved by localizing all the boundaries of an object. However, the full contour is hard to be identified in various cases, especially in the case of a liver on abdominal CT images. Figure 3.9. shows a gradient map of a sample axial slice. Edge responses are unclear on local boundaries and many false edges (i.e., responses on non-boundary regions) hinder the design of a robust algorithm. Especially in the portal phase, contrast-enhanced vessels show relatively high gradients that make an accurate boundary classification difficult (Fig. 3.9). Even with the strong capability of the deep neural network, it is difficult to classify the entire contour which has ambiguous regions. That is, a multi-task framework by simply adding ground-truth contour loss to the base network (i.e., segmentation) can make a severe contradiction on intermediate layers of a neural network. Thus, the proposed network avoids training the full contour features that are unnecessary. The proposed method guides (i.e., self-supervises) the neural network to learn sparse but essential contours that can be a great complementary feature that acts as implicit attention for the deep contexts.

From the base architecture of the aforementioned auto-context framework, the network is extended by training contour features. The contour weighting map that has larger values for the misclassified contour is first calculated:

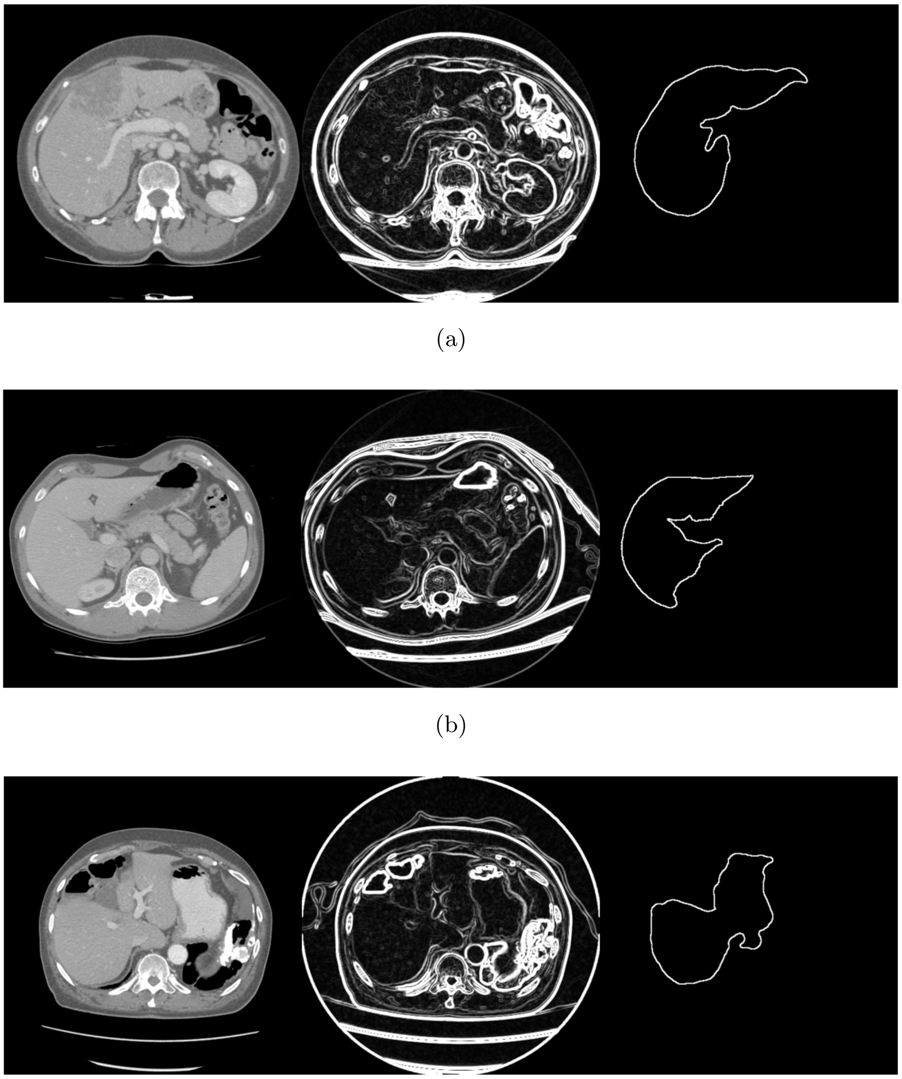
$$\widehat{\Gamma_c} = \Gamma_c \otimes \tilde{\mathbf{y}}_{\mathbf{l}}^{-1}, \tag{3.8}$$

where Γ_c, $\otimes$, and $\tilde{\mathbf{y}}_{\mathbf{l}}^{-1}$ indicate the ground-truth contour image, element-wise multiplication operator, and the final inverse liver prediction, respectively. The ground-truth contour image contains a value of 1 for the contour and 0 elsewhere. For the inverse prediction, $\tilde{y}_{l,i}^{-1} = 1 - \tilde{y}_{l,i}$ is applied for every i^{th} voxel where $\tilde{\mathbf{y}}_{\mathbf{l}}$ is the final output prediction of a foreground liver after softmax operation.

To employ penalization to the contour loss, the categorical cross-entropy classification loss is relaxed by a pixel-wise weighting scheme which is formed by the contour weighting map (3.8). The proposed method suggests a contour loss function as:

$$\mathrm{L}_{contour} = \psi(\tilde{\mathbf{y}}_{\mathbf{c}}, \Gamma_c, \widehat{\Gamma_c}) = -\sum_{i\in\Omega} \big(w_0(1-\Gamma_c(i))\log(1-\tilde{\mathbf{y}}_{\mathbf{c}}(i)) + w_1\Gamma_c(i)\widehat{\Gamma_c}(i)\log(\tilde{\mathbf{y}}_{\mathbf{c}}(i))\big), \tag{3.9}$$

where $\tilde{\mathbf{y}}_{\mathbf{c}}$ is the output prediction of the contour after softmax operation, w_c denotes class-specific weights for class c, and Ω indicates the dimensions of the image (i.e., $\Omega \in \mathbb{R}^3$). Consequently, the contour loss includes contour attention based on the final output (3.8) which is used to penalize the confident output of the network at each iteration. The difference between the proposed loss function and the focal loss [145] is that the proposed self-supervision is intended to

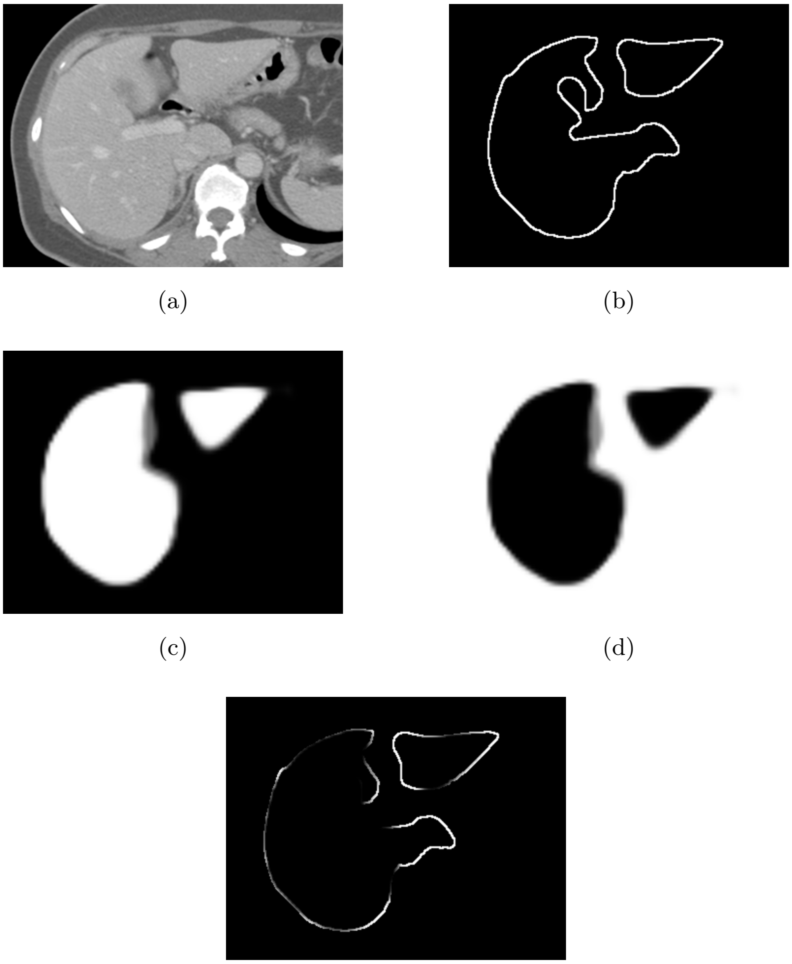


(a)

(b)

(c)

Figure 3.9: Example axial slices, corresponding edge responses (i.e., gradient magnitudes), and ground-truth contours. The first column shows example images, the second column illustrates gradient responses of each image, and the third column represents ground-truth contours.



(a)

(b)

(c)

(d)

(e)

Figure 3.10: Example of ground-truth contour and penalization map. The ground-truth contour is penalized by the prediction of the network. (a) An example axial slice of the original image. (b) A full ground-truth contour image (i.e., Γ_c). (c) The final prediction of the network (i.e., $\tilde{\mathbf{y}}_{\mathbf{l}}$). (d) The inverted liver prediction of (c) (i.e., $\tilde{\mathbf{y}}_{\mathbf{l}}^{-1}$). (e) A penalized contour image (i.e., $\widehat{\Gamma_c} = \Gamma_c \otimes \tilde{\mathbf{y}}_{\mathbf{l}}^{-1}$) for self-supervision (i.e., multiplication of (b) and (d)).

penalize (i.e., lower down) confident output regarding the final liver prediction rather than confidence of contour itself.

Figure 3.10 illustrates the generation process of the penalization map. A prediction of a liver (i.e., final output prediction of the network for each iteration; Fig. 3.10c) is inverted (Fig. 3.10d) and multiplied by the ground-truth contour image (Fig. 3.10b). Figure 3.10e shows the penalization map (i.e., $\widehat{\Gamma_c} = \Gamma_c \otimes \tilde{\mathbf{y}}_{\mathbf{l}}^{-1}$). The major significance of employing a penalization is that a confident region at boundaries is not enforced to be trained. A strong contour loss is mainly applied to the missing parts that can make the network to regularize feature extractor to give more attention to failed regions. The objective of giving contour loss is to guide the weights in intermediate feature layers rather than predict missing counterparts for further integration. Figure 3.11 illustrates example contour responses with and without penalization. By using self-supervising fashion, contour sub-network outputs very sparse predictions where the network needs to attend for further accuracy improvements (the second row in Fig. 3.11). Note that if the final prediction of the network reaches close to the ground-truth annotation, contour features become more sparse, and ideally, no responses are expected within contour sub-network.

The network may be seen as a multi-task learning framework. However, the network was not enforced to explicitly inference multiple tasks. The proposed network internally guides the weights to represent the object contour features without supervising the entire contour image. The network was self-supervised with penalization for each iteration. The main underlying principle of the proposed network is to concentrate the contour delineation on the missing contour part of an object (i.e., fine details of an object that are easily misclassified using the end-to-end learning). There are two main reasons for using the proposed method: 1) even with a powerful deep neural network, unclear boundaries are

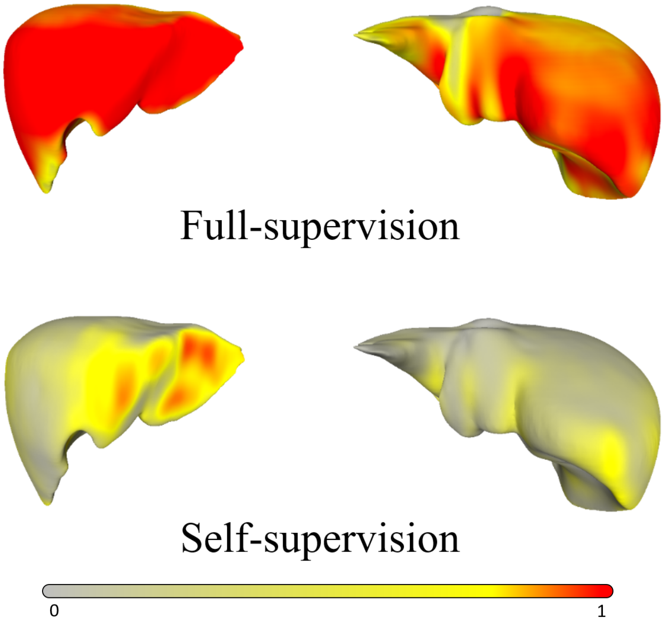


Figure 3.11: Example results of the contour network for the two different contour losses. The first row shows the contour responses based on the full ground-truth contour loss. The second row shows the contour responses based on self-supervising fashion. The self-supervision mechanism derives more sparse responses than the full-supervision.

challenging to be discriminated against as a contour and 2) contour regions in unclear boundaries can be compromised by global shapes. That is, the network avoids to learn complex boundary features that can be easily obtained by macroscopic shape features. The proposed network can be intuitively interpreted as a robust contour attention guided shape estimation.

3.4 Learning the network

3.4.1 Overall loss function

The task of a given learning system is to maximize the posterior, $p(\mathbf{y}|\mathbf{x})$. To effectively model the probability distribution, the proposed network model is trained to map the segmentation function $\phi(\mathbf{x}) : \mathbf{x} \longrightarrow \{0, 1\}$ by minimizing the following loss function:

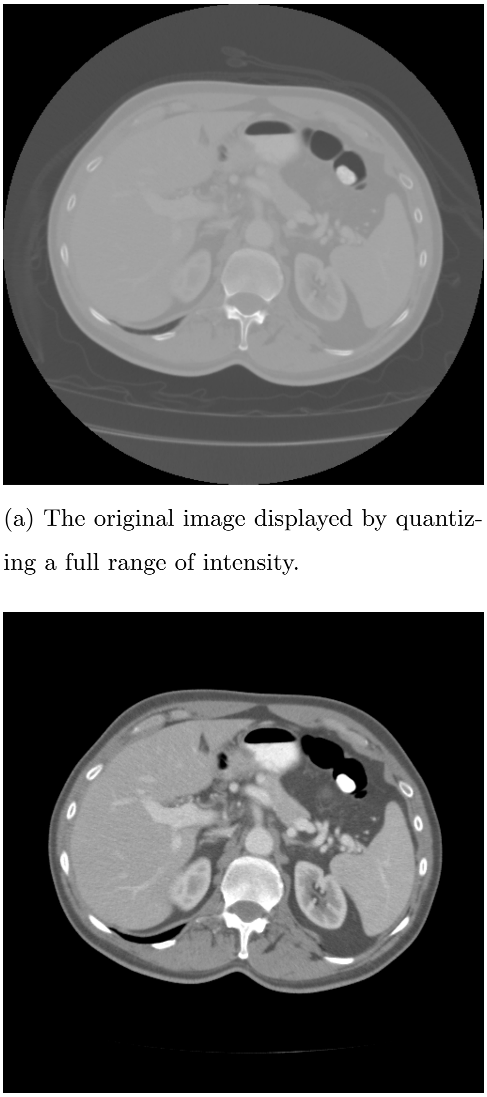
$$L_{autocenet} = L_{final} + \alpha L_{prior} + \beta L_{contour} + \gamma \|W\|_2^2, \tag{3.10}$$

where L_{final}, L_{prior}, and $L_{contour}$ indicate objective functions defined at the final layer (3.4), shape-prior layer (3.3), and contour layer (3.9), respectively. W is a whole set of network parameters. α, β, and γ in (3.10) are weighting parameters. The output of the network is obtained by applying softmax to the final output feature maps.

'Xavier' initialization [146] is used for initializing all the weights of the proposed network. While training the network, the loss parameters are fixed to $\alpha = \beta = 1$ and $\gamma = 0.001$ in (3.10). Adam optimizer [147] with batch size 4 and learning rate 0.001 were applied. The learning rate was decayed by multiplying 0.5 for every 10 epoch. The network was trained for 100 epochs using an Intel i9-7900X desktop system with 3.30 GHz processors, 128 GB of memory, and Nvidia Titan RTX (24 GB) GPU machine. The PyTorch framework was employed for the implementation of the network. It took 2h to complete all the training procedures.

3.4.2 Data augmentation

For the training dataset, all abdominal CT images were resampled into $128 \times 128 \times 64$. The images were pre-processed using fixed windowing values: level=10



(a) The original image displayed by quantizing a full range of intensity.

(b) Re-scaled image by quantizing fixed windowing values (i.e., level=10 and width=700).

Figure 3.12: Demonstration of windowing operation (intensity re-scaling). All training images are pre-processed by a given windowing operation (b).

and width=700 (i.e., clipped the intensity values under -340 and over 360) (Fig. 3.12). After re-scaling, the input images were normalized into the range [0..1] for each voxel. On-the-fly random affine deformation and additive noise were subsequently applied to the training images for each iteration with 80% probability.

An affine deformation is defined by randomly shearing an input image in the range of [-10..10] degrees for every three axes. In the case of noise addition, unit normal distribution of Gaussian noise (i.e., zero mean and $\sigma = 1$ standard deviation) is added. The affine and noise augmentations were individually applied with an 80% probability for each batch training.

3.5 Experimental results

3.5.1 Overview

The aim and scope of this experimental section are to provide strengths and weaknesses among the state-of-the-art CNN-based algorithms and the proposed method. The designed architectures of neural networks and the corresponding measures of several criteria (e.g., accuracy and generalization) are presented. Classical methods are not a scope of this study because it is a study of neural networks and it is also well-known that CNN-based methods are driving groundbreaking results in recent years.

A configuration of prepared data and metric of accuracy evaluation are first presented in the following sections followed by the overall comparison among the state-of-the-arts. Subsequently, a rich ablation study of the proposed method and rich visual analyses of intermediate features are investigated. Multiple N-fold cross-validations are thoroughly studied to provide an in-depth analysis of generalization performance over state-of-the-art networks and the proposed

Table 3.1

Training data configurations.

Data	Liver	Tumor	Phases	Subjects
Gibson et al. [63] (DenseVNet[1])	O	X	Multi	90
MICCAI-SLiver07 [82]	O	O	Portal	20
3Dircadb[2]	O	O	Portal	20
CHAOS challenge[3]	O	X	Portal	20
Ours	O	O/X	Portal	30

ablations. Finally, failure cases are demonstrated in the proposed method.

3.5.2 Data configurations and target of comparison.

In total, 180 subjects were acquired: 90 subjects from a publicly available dataset[1] presented in [63], 20 subjects from MICCAI-Sliver07 dataset [82], 20 subjects from 3Dircadb[2], 20 subjects from CHAOS challenge[3], and additional 30 annotated subjects with the help of clinical experts in the field. In the dataset, the slice thickness ranged from $0.5 - 5.0$mm and pixel sizes ranged from $0.6-1.0$mm. The dataset include multiple phases with and without tumors (Table 3.1).

In the experiments, the performance of accuracy and generalization of the proposed network is evaluated by comparing them with those of the other state-of-the-art FCN-based models. The state-of-the-art networks, 3D U-Net [58], V-Net [60], deeply supervised network (DSN) [26], voxel-wise residual network (VoxResNet) [62], Dense V-Network (DenseVNet) [63], attention gated U-Net

[1] https://doi.org/10.5281/zenodo.1169361
[2] https://www.ircad.fr/research/3dircadb
[3] https://doi.org/10.5281/zenodo.3367758

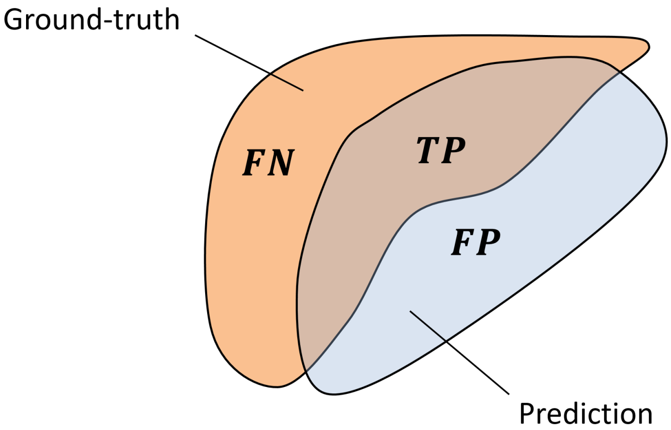


Figure 3.13: An illustration of true positives, false positives, and false negatives.

(AGU-Net) [52], and the proposed network, AutoCENet are used for the performance evaluation.

3.5.3 Evaluation metric

The results of segmentation were evaluated using the dice similarity coefficient (DSC), precision, sensitivity, Hausdorff distance (HD), and average symmetric surface distance (ASSD). The DSC is defined as follows:

$$DSC(X,Y) = \frac{2|X \cap Y|}{|X| + |Y|}, \tag{3.11}$$

where $|\cdot|$ is the cardinality of a set. Precision and sensitivity are defined by

$$Precision = \frac{TP}{TP + FP} \tag{3.12}$$

$$Sensitivity = \frac{TP}{TP + FN}, \tag{3.13}$$

where TP, FN, and FP are the numbers of true positive, false negative, and false positive voxels, respectively (Fig. 3.13). The remaining surface distance metrics are evaluated on a surface basis. Let $\mathbf{S}_X$ be a set of surface voxels of a set X, the shortest distance of an arbitrary voxel p is defined as [82]:

$$d(p, \mathbf{S}_X) = \min_{s_X \in \mathbf{S}_X} ||p - s_X||_2. \tag{3.14}$$

HD is thus given by [82]:

$$HD(X, Y) = \max\{\max_{s_X \in \mathbf{S}_X} d(s_X, \mathbf{S}_Y) + \max_{s_Y \in \mathbf{S}_Y} d(s_Y, \mathbf{S}_X)\}. \tag{3.15}$$

Defining the distance function as

$$D(\mathbf{S}_X, \mathbf{S}_Y) = \sum_{s_X \in \mathbf{S}_X} d(s_X, \mathbf{S}_Y), \tag{3.16}$$

the ASSD can be defined as [82]:

$$ASSD(X, Y) = \frac{1}{|\mathbf{S}_X| + |\mathbf{S}_Y|}(D(\mathbf{S}_X, \mathbf{S}_Y) + D(\mathbf{S}_Y, \mathbf{S}_X)). \tag{3.17}$$

In (3.15), 95% of voxels were additionally calculated in (3.14) to exclude 5% outlying voxels. 95% HD can be a better generalized evaluation of distance because there exists ground-truth variations on portal vein region.

For the fair comparison among the networks, the same data augmentations were applied for every network such as resampling of the resolution, rescaling of intensities, and affine deformations (refer to section 3.4.2). The CRF post-processing presented in [26] was also excluded for the same respect. The auto-context algorithm of VoxResNet [62] is adopted to compare the algorithms.

All hyperparameters were set as specified in Table 3.2. The performance of training for each network had no penalty by unifying the hyperparameters. That is, there was no significant difference by setting the hyperparameters (e.g., batch size, optimization metric, and learning rate) as specified in the original papers (while a few improvements have been monitored by using configurations presented in Table. 3.2). The dataset was composed of three separate sets of training, validation, and testing (Table 3.3). The dataset was firstly randomly shuffled, and 100 images were used for two-fold cross-validation and 80 images were used for all the accuracy evaluation (i.e., testing).

Table 3.2
Hyperparameters and metrics used in training.

Parameters	Value or Metric
Optimizer	Adam [147]
Learning rate	0.001
Learning rate decay per epoch (decay/epoch)	0.5 / 10
Weight decay (L_2 regularization)	0.001

Table 3.3
Number of the training datasets.

Total	Training	Validation	Testing
180	50	50	80

3.5.4 Accuracy evaluation

The results in Table 3.4 show that the proposed AutoCENet outperformed other state-of-the-arts. The AutoCENet showed the highest values in DSC, precision, and sensitivity and showed the lowest values in HD, 95% HD, and ASSD that indicate the proposed network presented the best results. Moreover, the proposed network showed better accuracy while using much fewer parameters than the other state-of-the-art methods. The lowest precision and sensitivity were presented by DenseVNet [63] indicating that the results contained severe false positives and false negatives. The DenseVNet failed to segment the liver accurately due to the two significant reasons: the resolution of the network is too low and shape prior has a weak representative power. The excessively coarse dimensions of the network suffer from the accurate segmentation in the original image resolution. Furthermore, 12^3 resolution of shape-prior is too small and training images must be accurately, and manually cropped for the robustness of the shape-prior. There is no specific metric presented in the original work [63]

Table 3.4
Accuracy evaluation of the proposed network and other state-of-the-arts.

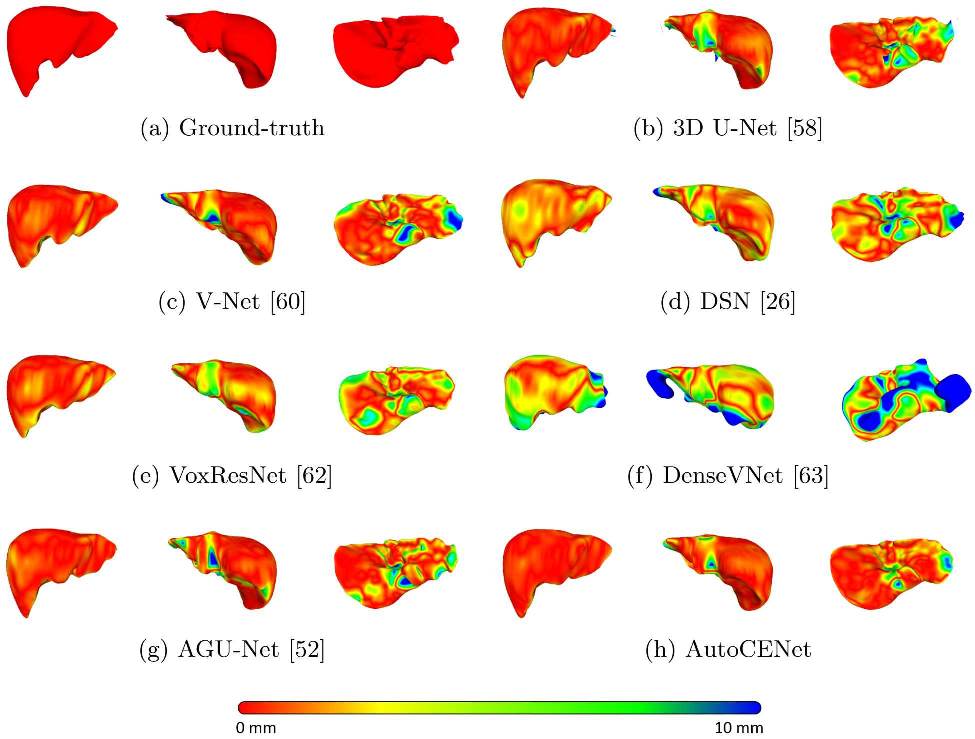
Methods	DSC	Precision	Sensitivity	HD [mm]	95% HD [mm]	ASSD [mm]
3D U-Net	0.95 ± 0.01	0.94 ± 0.02	0.96 ± 0.02	45.20±31.93	7.77 ± 12.71	1.33 ± 0.91
V-Net	0.95 ± 0.02	0.94 ± 0.02	0.95 ± 0.03	26.52±19.05	5.38 ± 3.94	1.20 ± 0.65
DSN	0.92 ± 0.02	0.88 ± 0.04	0.97 ± 0.01	28.63±23.85	7.40 ± 9.33	1.77 ± 1.05
VoxResNet	0.95 ± 0.01	0.95 ± 0.02	0.95 ± 0.02	18.67±11.15	4.99 ± 5.89	1.11 ± 0.49
DenseVNet	0.83 ± 0.05	0.75 ± 0.09	0.94 ± 0.03	37.19±14.52	16.54 ± 8.47	3.98 ± 1.69
AGU-Net	0.95 ± 0.01	0.94 ± 0.03	0.96 ± 0.01	31.57±22.22	8.56 ± 13.52	1.34 ± 1.07
AutoCENet	**0.96 ± 0.01**	**0.95 ± 0.02**	**0.97 ± 0.01**	**14.96±4.25**	**2.92 ± 1.12**	**0.82 ± 0.32**

Table 3.5
Accuracy evaluation of the proposed network and other state-of-the-arts (CCA post-processing).

Methods	DSC	Precision	Sensitivity	HD [mm]	95% HD [mm]	ASSD [mm]
3D U-Net	0.95 ± 0.01	0.95 ± 0.02	0.96 ± 0.02	16.86 ± 4.78	4.23 ± 1.53	1.00 ± 0.38
V-Net	0.95 ± 0.02	0.94 ± 0.02	0.95 ± 0.02	20.51 ± 9.31	5.07 ± 3.23	1.18 ± 0.60
DSN	0.92 ± 0.02	0.88 ± 0.03	0.97 ± 0.01	16.49 ± 4.50	5.31 ± 2.13	1.56 ± 0.66
VoxResNet	0.95 ± 0.01	0.95 ± 0.02	0.95 ± 0.02	16.86 ± 8.12	4.07 ± 1.51	1.08 ± 0.41
DenseVNet	0.83 ± 0.05	0.75 ± 0.08	0.94 ± 0.03	32.21±12.76	16.03 ± 8.04	3.90 ± 1.63
AGU-Net	0.95 ± 0.01	0.95 ± 0.02	0.96 ± 0.01	18.77 ± 8.14	4.72 ± 2.85	1.04 ± 0.43
AutoCENet	**0.95 ± 0.01**	**0.95 ± 0.02**	**0.97 ± 0.01**	**14.95±4.24**	**2.91 ± 1.09**	**0.82 ± 0.32**

to crop testing images automatically. Example visualizations of the results are visualized in Figs. 3.14 and 3.15. All visualized surfaces are smoothed by the curvature flow smoothing method [148] at the original image resolution.

To eliminate false responses of the networks, connected component analysis (CCA) [149] is performed as a post-processing procedure. The performance of the networks with CCA is demonstrated in Table 3.5. Significant improvements were found in other state-of-the-art networks regarding the distance metrics. The HD and 95% HD values were lowered in a huge margin except for the proposed AutoCENet which indicates that the AutoCENet showed little false responses. Figure 3.16 and 3.17 demonstrate box plots of Tables 3.4 and 3.5. The presented box plots show a significant reduction of distances by applying



(a) Ground-truth

(b) 3D U-Net [58]

(c) V-Net [60]

(d) DSN [26]

(e) VoxResNet [62]

(f) DenseVNet [63]

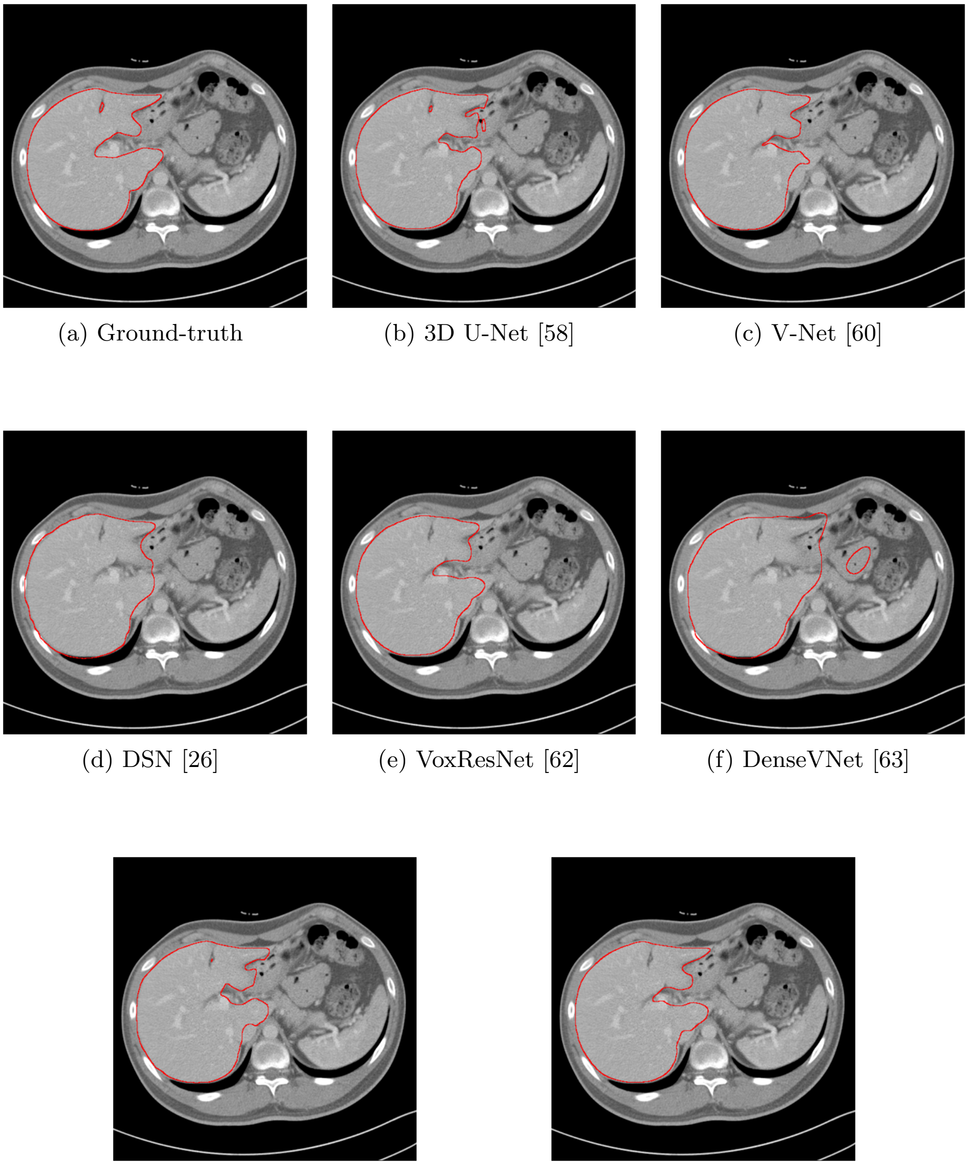
(g) AGU-Net [52]

(h) AutoCENet

Figure 3.14: Example visualizations of the test results for state-of-the-art networks. The surface color is visualized based on the distance to the ground-truth surface. Visualized surfaces are smoothed by the curvature flow smoothing method [148] at the original image resolution.

CCA to the final output results.

The DSN [26] in Table 3.4 showed high ASSD because the inference of the network are made from low resolution. The up-sampling from $40 \times 40 \times 18$ has limitations to accurately delineate objects in the original resolution. Furthermore, even with low-dimensional representation, DSN showed false positives indicating that several deep supervisions did not successfully achieve the discrimination of the high-level features. In fact, multiple deep supervision enforces the lower-level intermediate features to be discriminative that resulted in degra-



(a) Ground-truth

(b) 3D U-Net [58]

(c) V-Net [60]

(d) DSN [26]

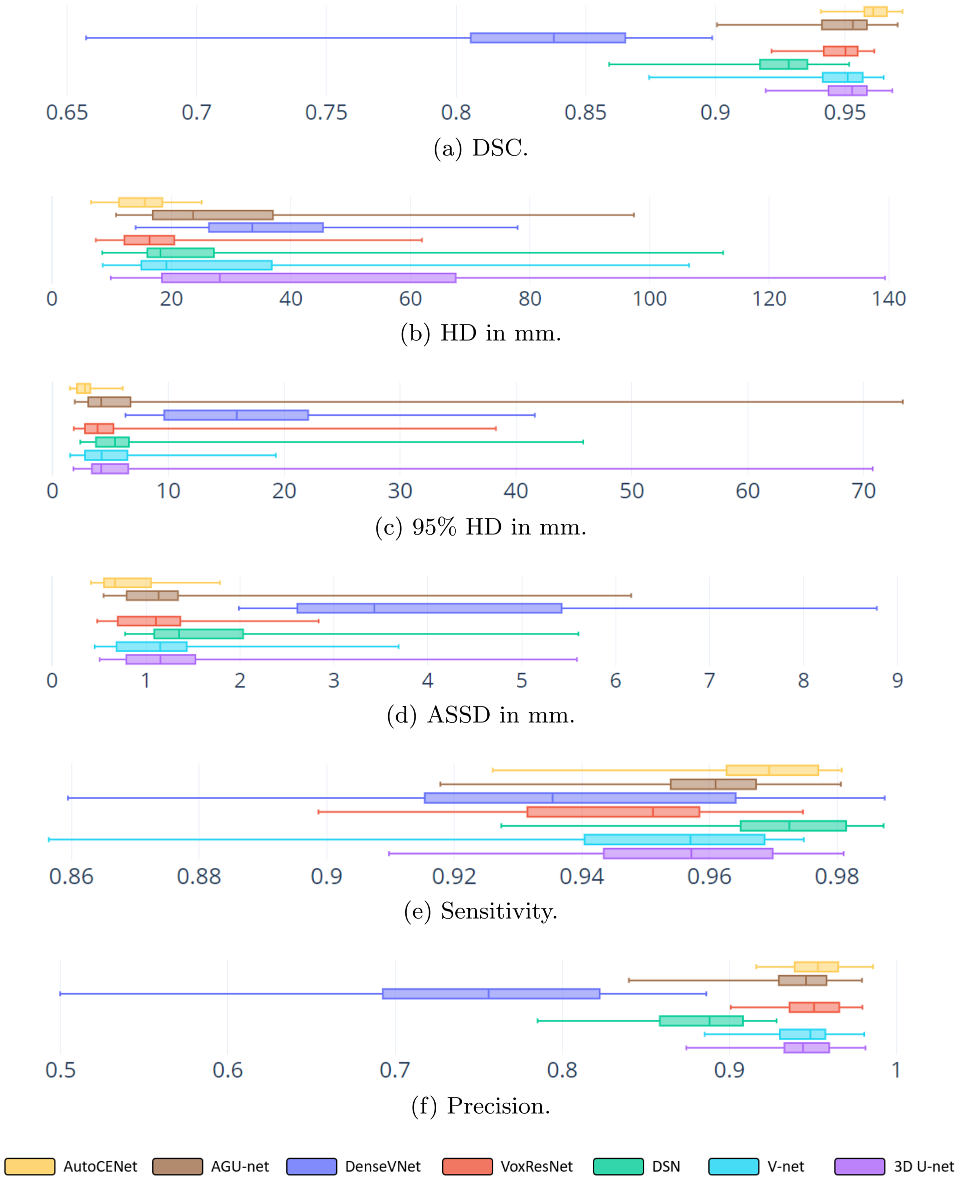
(e) VoxResNet [62]

(f) DenseVNet [63]

(g) AGU-Net [52]

(h) AutoCENet

Figure 3.15: Example axial slices of the test results for state-of-the-art networks.



(a) DSC.

(b) HD in mm.

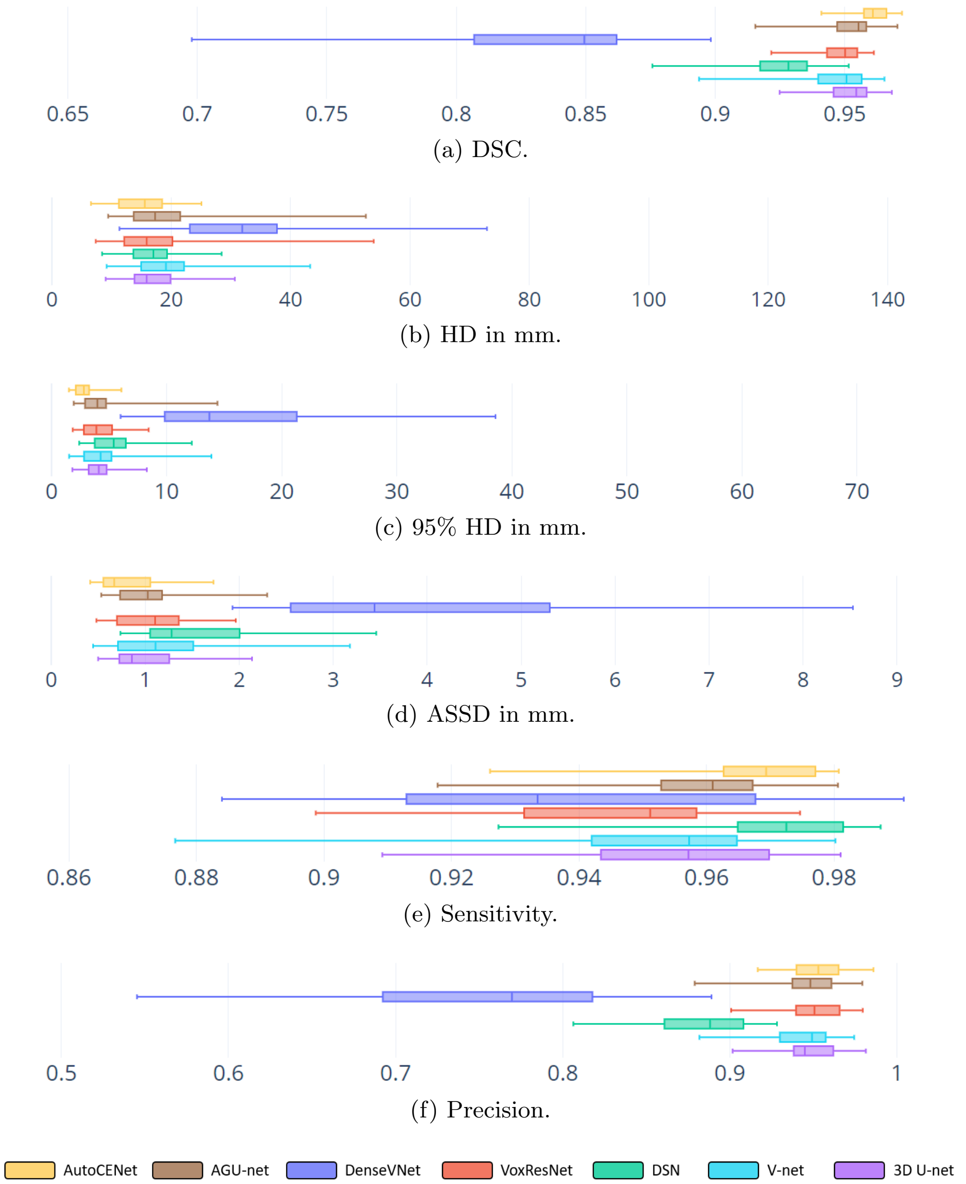
(c) 95% HD in mm.

(d) ASSD in mm.

(e) Sensitivity.

(f) Precision.

Figure 3.16: Box plots of the evaluation metrics for state-of-the-arts.



(a) DSC.

(b) HD in mm.

(c) 95% HD in mm.

(d) ASSD in mm.

(e) Sensitivity.

(f) Precision.

Figure 3.17: Box plots of the evaluation metrics for state-of-the-arts (CCA post-processing).

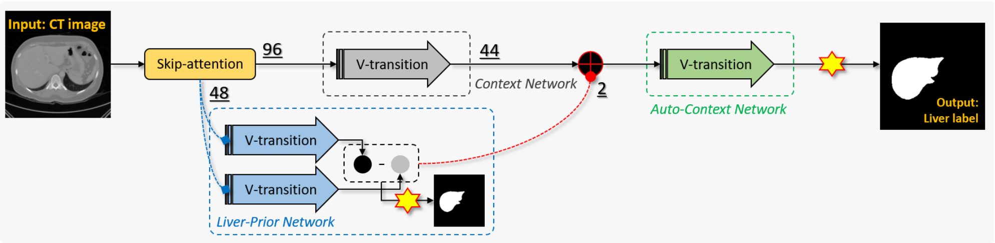


Figure 3.18: AutoNet baseline (without contour transition sub-network).

dation of overall performance. The AGU-Net also presented false positives as opposed to the architectural design principle proposed in the reference [52]. The spatial attention gated units in AGU-Net [52] failed to suppress irrelevant background regions as suggested. On the other hand, VoxResNet [62] showed the second minimum distance gaps between using and not using the CCA post-processing (Tables 3.4 and 3.5). The small distance gap indicates the auto-context algorithm, employed to the VoxResNet, successfully suppressed false positive responses.

3.5.5 Ablation study

In this section, the ablations of the proposed AutoCENet are studied to verify the architectural components of the network. To justify the designed methods thoroughly, the study contains two major categories: auto-context architecture and contour loss. The first subsection covers a few ablations regarding the proposed auto-context framework without contour loss. The second subsection demonstrates several auxiliary contour-related losses to elucidate the supremacy of the proposed self-supervised contour loss.

Auto-context framework

The auto-context framework is validated which does not exploit self-supervising contours (i.e., without contour loss, $L_{contour}$ in (3.10); Fig. 3.18). From the

Table 3.6
Accuracy evaluation of the proposed network and auto-context ablations.

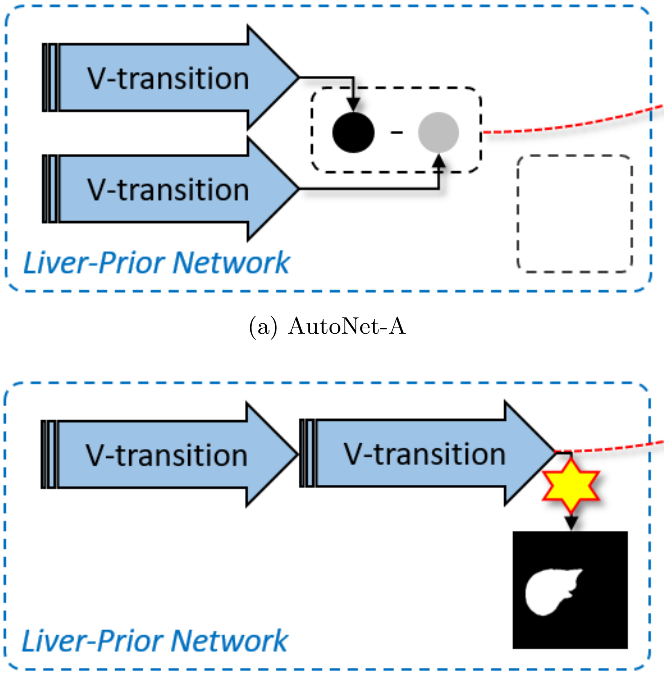
Methods	DSC	Precision	Sensitivity	HD [mm]	95% HD [mm]	ASSD [mm]
AutoCENet	**0.96 ± 0.01**	**0.95 ± 0.02**	**0.97 ± 0.01**	**14.96±4.25**	**2.92 ± 1.12**	**0.82 ± 0.32**
AutoNet	0.95 ± 0.01	0.95 ± 0.02	0.96 ± 0.02	20.18 ± 8.79	4.48 ± 2.47	1.04 ± 0.42
AutoNet-att	0.95 ± 0.01	0.95 ± 0.02	0.96 ± 0.02	25.73±17.06	5.83 ± 5.93	1.10 ± 0.53
AutoNet-A	0.95 ± 0.01	0.94 ± 0.02	0.95 ± 0.03	33.25±22.80	7.61 ± 8.75	1.34 ± 0.71
AutoNet-R	0.95 ± 0.01	0.94 ± 0.02	0.95 ± 0.02	37.99±25.09	5.46 ± 3.61	1.23 ± 0.57
AutoNet-AR	0.94 ± 0.01	0.94 ± 0.02	0.95 ± 0.03	38.88±28.81	6.24 ± 4.67	1.32 ± 0.61
AutoNet+P	0.95 ± 0.01	0.95 ± 0.02	0.96 ± 0.02	20.70±12.59	4.09 ± 2.66	0.99 ± 0.52

Table 3.7
Accuracy evaluation of the proposed network and auto-context ablations (CCA post-processing).

Methods	DSC	Precision	Sensitivity	HD [mm]	95% HD [mm]	ASSD [mm]
AutoCENet	**0.96 ± 0.01**	**0.95 ± 0.02**	**0.97 ± 0.01**	**14.96±4.25**	**2.92 ± 1.12**	**0.82 ± 0.32**
AutoNet	0.95 ± 0.01	0.95 ± 0.02	0.96 ± 0.02	19.76 ± 8.70	4.51 ± 2.49	1.04 ± 0.42
AutoNet-att	0.95 ± 0.01	0.95 ± 0.02	0.96 ± 0.02	20.44±10.26	4.91 ± 3.66	1.04 ± 0.45
AutoNet-A	0.95 ± 0.01	0.94 ± 0.02	0.95 ± 0.03	22.40±14.11	6.44 ± 6.33	1.23 ± 0.64
AutoNet-R	0.95 ± 0.01	0.94 ± 0.02	0.95 ± 0.02	21.89 ± 9.94	5.27 ± 3.44	1.16 ± 0.50
AutoNet-AR	0.94 ± 0.01	0.94 ± 0.02	0.95 ± 0.03	22.41±11.80	6.14 ± 4.61	1.25 ± 0.56
AutoNet+P	0.96 ± 0.01	0.95 ± 0.02	0.96 ± 0.02	17.51 ± 7.19	3.96 ± 2.44	0.95 ± 0.43

base auto-context framework the three additional ablations are studied: the one without auto-context part (i.e., AutoNet-A; Fig. 3.19a), without high-level residual inference (i.e., AutoNet-R; Fig. 3.19b), and without them both (i.e., AutoNet-AR). For AutoNet-A, the deep supervision for the liver-prior network (i.e., L_{prior} in (3.10) is removed. In the case of AutoNet-R, the high-level residual connection is modified to a sequential connection of V-transitions with the intermediate number of features by 48. AutoNet-AR employed both modifications of corresponding AutoNet-A and AutoNet-R. Additionally, the removal of base channel-wise attention (i.e., AutoNet-att) and addition of confident penalization of the output (i.e., AutoNet+P) was evaluated.

The results (Table 3.6) show that the accuracy of all the ablations was lower

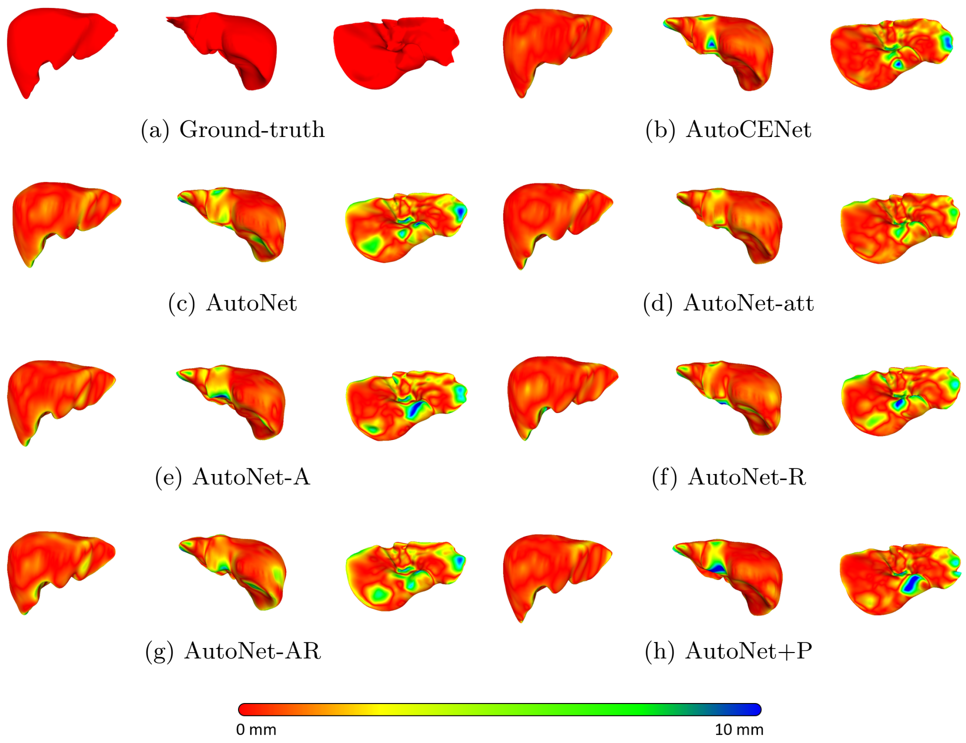


(a) AutoNet-A

(b) AutoNet-R

Figure 3.19: Ablation of AutoNet. (a) AutoNet without auto-context algorithm. (b) AutoNet without residual connection.

than the original AutoCENet. Comparing with AutoNet, the ablations showed lower performances except for AutoNet+P. The penalization of confident output (2.9) from AutoNet baseline showed comparable results. The other ablations showed a significant increase regarding distance metrics by not employing the auto-context algorithm or residual shape-prior. The results indicate that the auto-context framework and residual shape-prior estimation jointly play an important role in the final accuracy. The results of the liver-prior network with and without the residual showed that the high-level residual connection boosts the performance of a liver prior. Example visualizations of the results for AutoNet ablations are visualized in Figs. 3.20 and 3.21.



(a) Ground-truth

(b) AutoCENet

(c) AutoNet

(d) AutoNet-att

(e) AutoNet-A

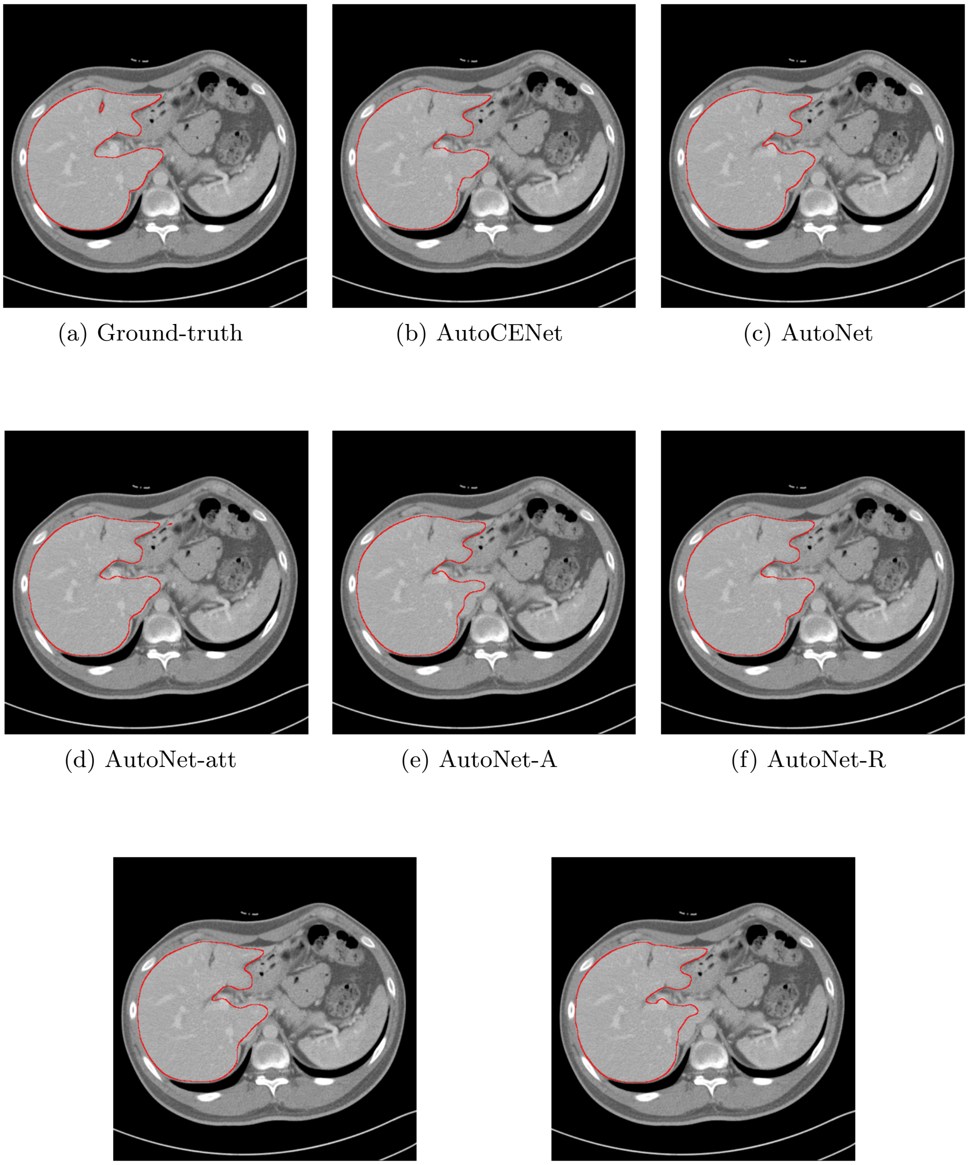
(f) AutoNet-R

(g) AutoNet-AR

(h) AutoNet+P

Figure 3.20: Example visualizations of the test results for AutoNet and the corresponding ablations. The surface color is visualized based on the distance to the ground-truth surface. Visualized surfaces are smoothed by the curvature flow smoothing method [148] at the original image resolution.

Table 3.7 presents the results by applying CCA post-processing from the Table 3.6. The major indication from the two tables is that the ablation of auto-context algorithm and residual estimation results in severe false responses on the final output predictions. Figures 3.22 and 3.23 demonstrate box plots of Tables 3.6 and 3.7. The AutoNet showed little improvement of distances by applying CCA indicating that a complete, proposed auto-context algorithm showed no significant false responses.



(a) Ground-truth

(b) AutoCENet

(c) AutoNet

(d) AutoNet-att

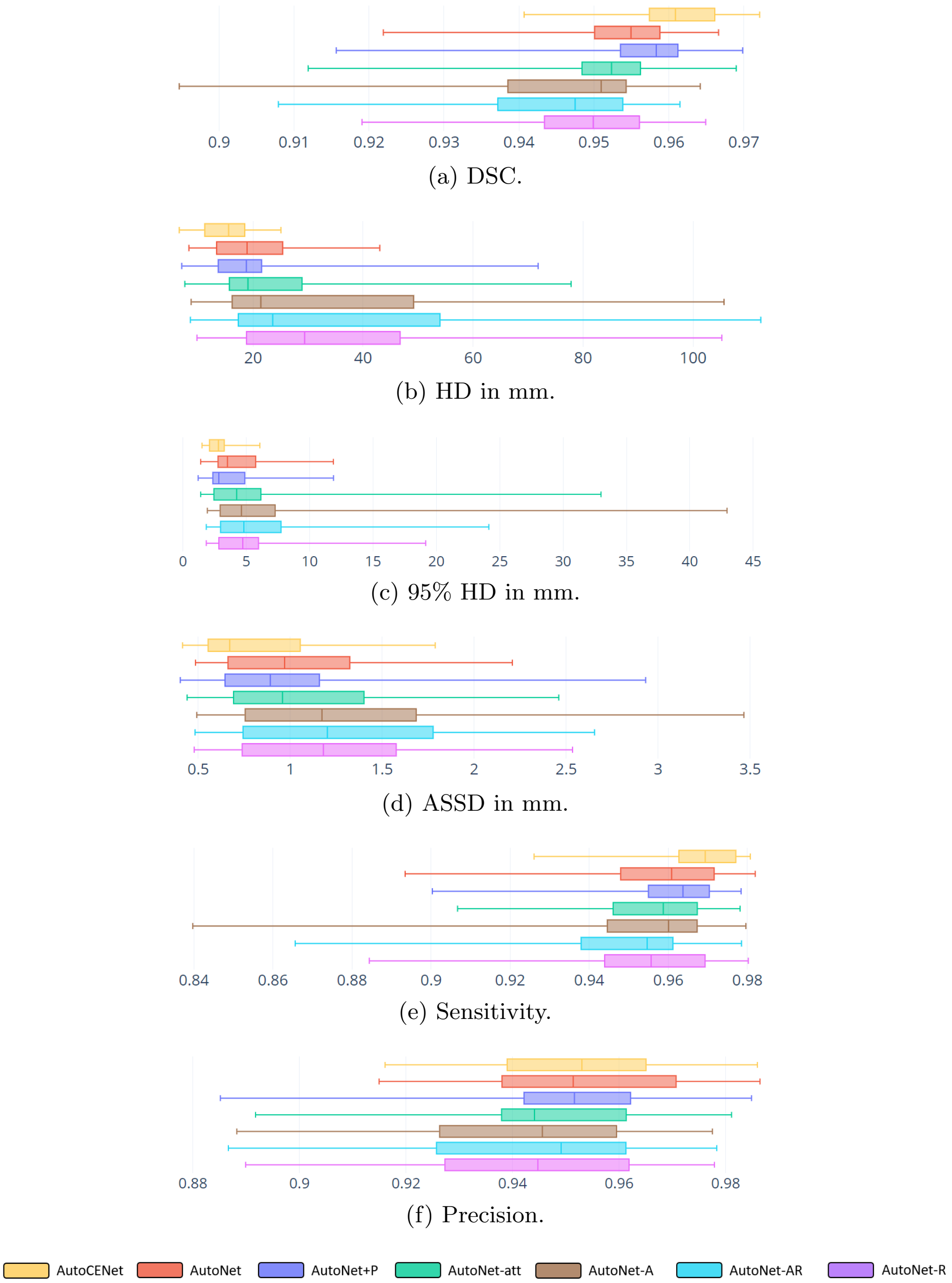
(e) AutoNet-A

(f) AutoNet-R

(g) AutoNet-AR

(h) AutoNet+P

Figure 3.21: Example axial slices of the test results for AutoNet and the corresponding ablations.



(a) DSC.

(b) HD in mm.

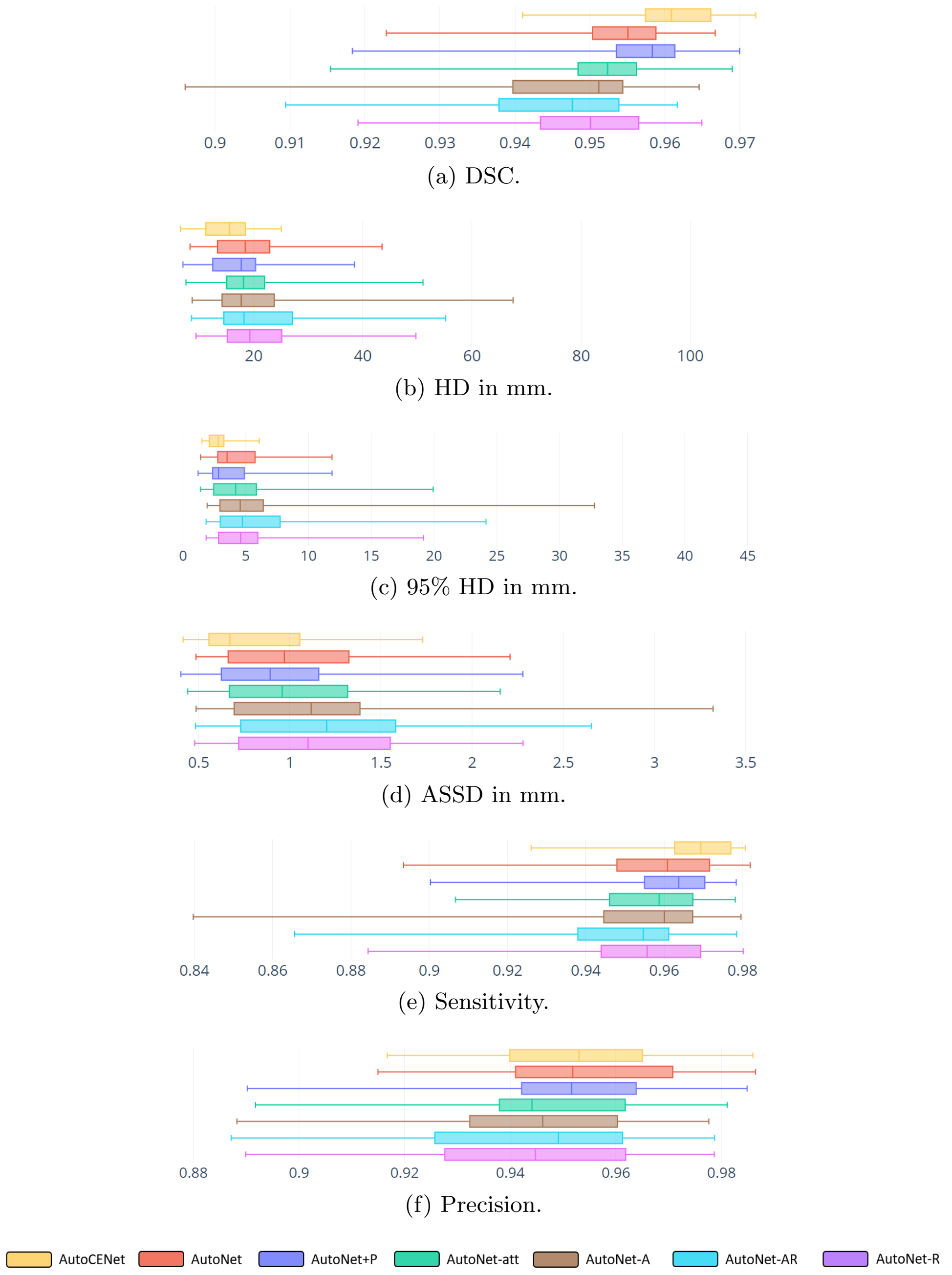
(c) 95% HD in mm.

(d) ASSD in mm.

(e) Sensitivity.

(f) Precision.

Figure 3.22: Box plots of the evaluation metrics for auto-context ablations.



(a) DSC.

(b) HD in mm.

(c) 95% HD in mm.

(d) ASSD in mm.

(e) Sensitivity.

(f) Precision.

Figure 3.23: Box plots of the evaluation metrics for auto-context ablations (CCA post-processing).

Contour losses

For the ablation study of a contour loss, networks were designed by using the full-supervision for the contour loss (AutoCENet+FC), full-supervision with penalizing confident outputs of contour features (AutoCENet+FC+P), and a special penalization of auxiliary contour loss (AutoCENet+SP). The contour loss functions for each ablation is presented as follows. Full contour supervision as

$$\psi_{+FC}(\tilde{\mathbf{y}}_{\mathbf{c}}, \Gamma_c) = \chi(\tilde{\mathbf{y}}_{\mathbf{c}}, \Gamma_c) \tag{3.18}$$

and full-supervision with penalizing confident outputs as

$$\psi_{+FC+P}(\tilde{\mathbf{y}}_{\mathbf{c}}, \Gamma_c) = \chi(\tilde{\mathbf{y}}_{\mathbf{c}}, \Gamma_c) - \beta H(p(\tilde{\mathbf{y}}_{\mathbf{c}}|\mathbf{x};\theta)) \tag{3.19}$$

where $\tilde{\mathbf{y}}_{\mathbf{c}}$ is the output prediction of contour and χ indicates the cross-entropy loss. A special penalization is defined by

$$\psi_{+SP}(\tilde{\mathbf{y}}_{\mathbf{c}}, \mathbf{y}_{\mathbf{l}}, \widehat{\mathbf{y}_{\mathbf{l}}}) = \psi(\tilde{\mathbf{y}}_{\mathbf{c}}, \mathbf{y}_{\mathbf{l}}, \widehat{\mathbf{y}_{\mathbf{l}}}) \tag{3.20}$$

where ψ is same as defined in (3.9) and $\widehat{\mathbf{y}_{\mathbf{l}}}$ is defined as

$$\widehat{\mathbf{y}_{\mathbf{l}}} = \mathbf{y}_{\mathbf{l}} \otimes \tilde{\mathbf{y}}_{\mathbf{l}}^{-1}, \tag{3.21}$$

similar to (3.8). A special penalization defined in (3.20) trains the contour branch to delineate the misclassified regions on the final output prediction rather than the contour. The special penalization (SP) is designed to elucidate the difficulty of training complementary features and verify the proposed contour loss. The entropy function H in (3.19) is defined by

$$H(p(\mathbf{y}|\mathbf{x};\theta)) = -\sum_i p_\theta(\mathbf{y}_i|\mathbf{x})\log(p_\theta(\mathbf{y}_i|\mathbf{x})), \tag{3.22}$$

Table 3.8
Accuracy evaluation of the proposed network and the contour variants.

Methods	DSC	Precision	Sensitivity	HD [mm]	95% HD [mm]	ASSD [mm]
AutoCENet	**0.96 ± 0.01**	**0.95 ± 0.02**	**0.97 ± 0.01**	**14.96±4.25**	**2.92 ± 1.12**	**0.82 ± 0.32**
+P	0.96 ± 0.01	0.95 ± 0.02	0.96 ± 0.01	20.75±11.05	3.70 ± 1.73	0.93 ± 0.40
+FC	0.95 ± 0.01	0.94 ± 0.02	0.95 ± 0.02	27.56±20.34	5.66 ± 5.22	1.20 ± 0.56
+FC+P	0.95 ± 0.01	0.94 ± 0.03	0.96 ± 0.02	32.93±24.35	5.57 ± 4.11	1.20 ± 0.56
+SP	0.95 ± 0.01	0.95 ± 0.02	0.96 ± 0.02	29.30±21.48	4.20 ± 2.03	1.03 ± 0.45

Table 3.9
Accuracy evaluation of the proposed network and the contour variants (CCA post-processing).

Methods	DSC	Precision	Sensitivity	HD [mm]	95% HD [mm]	ASSD [mm]
AutoCENet	**0.96 ± 0.01**	**0.95 ± 0.02**	**0.97 ± 0.01**	**14.96±4.25**	**2.92 ± 1.12**	**0.82 ± 0.32**
+P	0.96 ± 0.01	0.95 ± 0.02	0.96 ± 0.01	17.16 ± 5.55	3.51 ± 1.39	0.90 ± 0.34
+FC	0.95 ± 0.01	0.95 ± 0.02	0.95 ± 0.02	21.12±11.42	5.63 ± 5.21	1.16 ± 0.54
+FC+P	0.95 ± 0.01	0.94 ± 0.02	0.96 ± 0.02	21.23±11.48	5.41 ± 3.98	1.13 ± 0.51
+SP	0.95 ± 0.01	0.95 ± 0.02	0.96 ± 0.02	18.53 ± 6.91	4.19 ± 1.99	0.90 ± 0.34

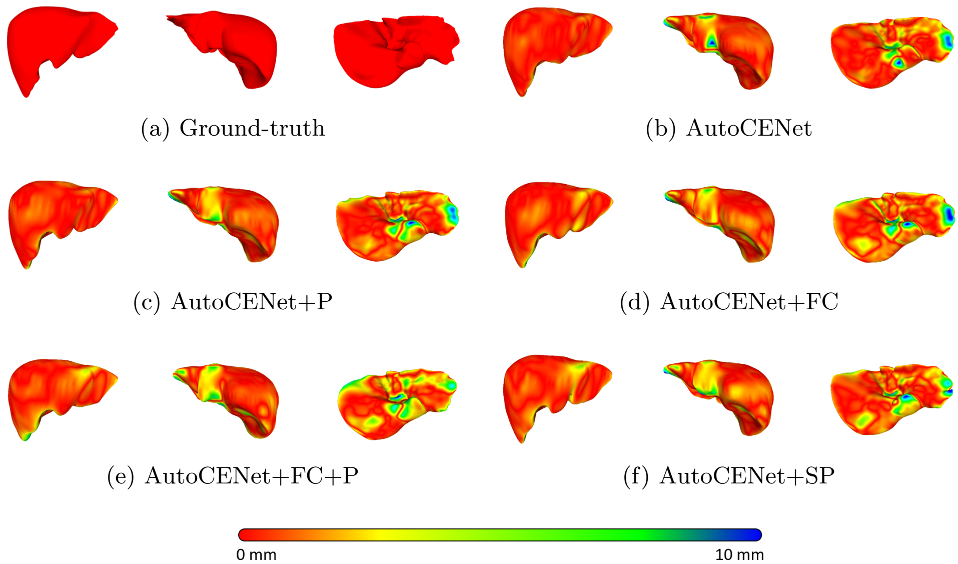
similar to [8]. β is a control parameter which defined the strength of the confidence penalty [8]. The remaining loss terms are the same as presented in (3.10).

Additionally, a global penalization of the final output of AutoCENet (AutoCENet+P) is presented. A global penalization is defined by

$$L_{autocenet+P}(\tilde{\mathbf{y}}_{\mathbf{l}}, \tilde{\mathbf{y}}_{\mathbf{c}}, \Gamma_c, \widehat{\Gamma_c}) = L_{contour} - \beta H(p(\tilde{\mathbf{y}}_{\mathbf{l}}|\mathbf{x};\theta)) \tag{3.23}$$

where $\tilde{\mathbf{y}}_{\mathbf{l}}$ is the final output prediction of liver and $L_{contour}$ is the same function as (3.9). The entropy function H is same as (3.22). Table 3.8 shows the accuracy assessment results of the contour variants.

All the contour variants showed lower accuracy compared to the original network. The performance of the AutoCENet+FC was even poorer than that of the AutoNet (Table 3.6) for the distance measures, indicating that enforcing the network to learn the full ground-truth contour image degrades the performance. Both penalization schemes on contour and global showed no improvements.


(a) Ground-truth

(b) AutoCENet

(c) AutoCENet+P

(d) AutoCENet+FC

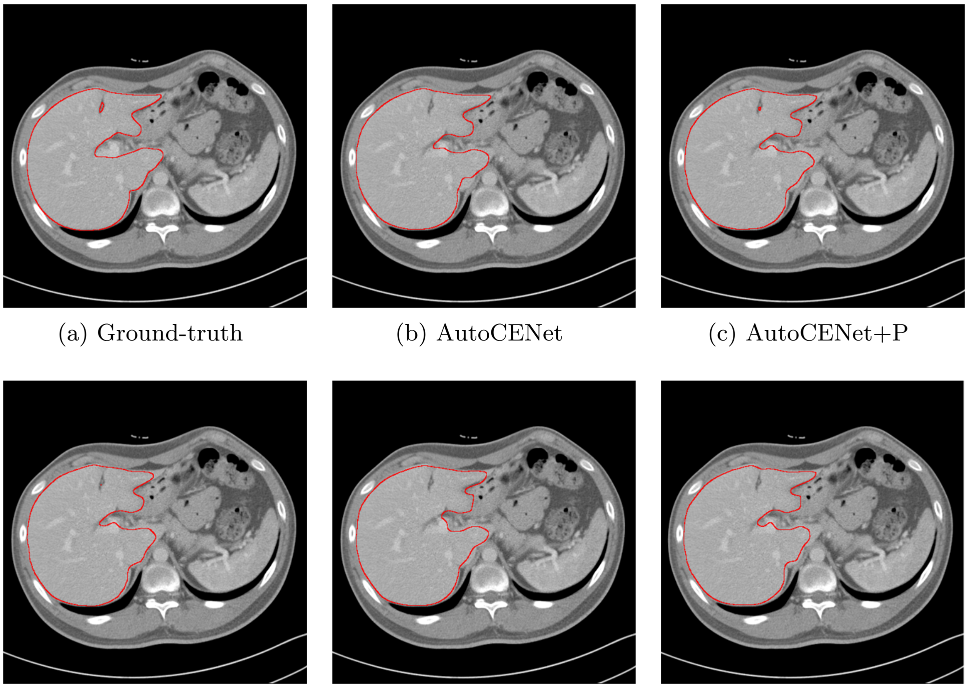
(e) AutoCENet+FC+P

(f) AutoCENet+SP

Figure 3.24: Example visualizations of the test results for contour variants. The surface color is visualized based on the distance to the ground-truth surface. Visualized surfaces are smoothed by the curvature flow smoothing method [148] at the original image resolution.

The special penalized training of misclassified regions for contour loss showed better performance than full-contour-based networks but was worse than that of AutoCENet. Example visualizations of the results for AutoNet ablations are visualized in Figs. 3.24 and 3.25.

Table 3.9 presents the results by applying CCA post-processing from the Table 3.8. Figures 3.26 and 3.27 demonstrate box plots of Tables 3.8 and 3.9. The ablations including full-contour supervisions and special penalization resulted in degradation of performance. The ablations also increased false responses on the final outputs.



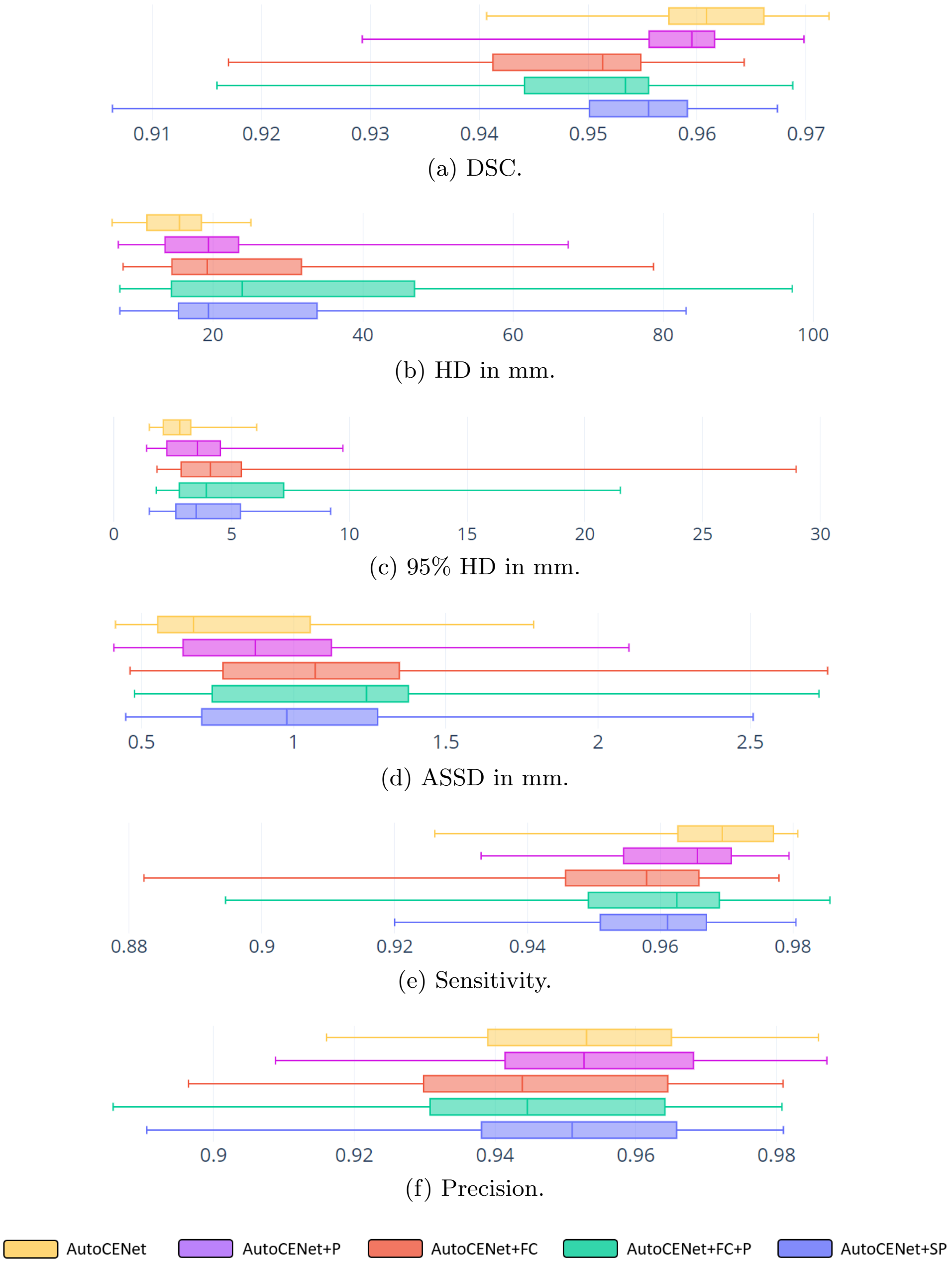
(a) Ground-truth (b) AutoCENet (c) AutoCENet+P

(d) AutoCENet+FC (e) AutoCENet+FC+P (f) AutoCENet+SP

Figure 3.25: Example axial slices of the test results for contour variants.

Shape-prior feature layer

Figures 3.28 and 3.29 show the liver shape-priors that are estimated with and without the residual connection. The predicted probabilities clearly show that the effectiveness of the high-level residual connection in shape prior estimation. The posterior of liver from AutoNet-R (Fig. 3.28f) shows significant false positive responses compared to the residual version (Fig. 3.28e). The two high-level predictions, i.e., Figs. 3.28c and 3.28d, were used as mutual complements to derive accurate liver prediction (3.28e). The results indicate that high-level residual inference shows an effective way to estimate accurate prior of liver region.



(a) DSC.

(b) HD in mm.

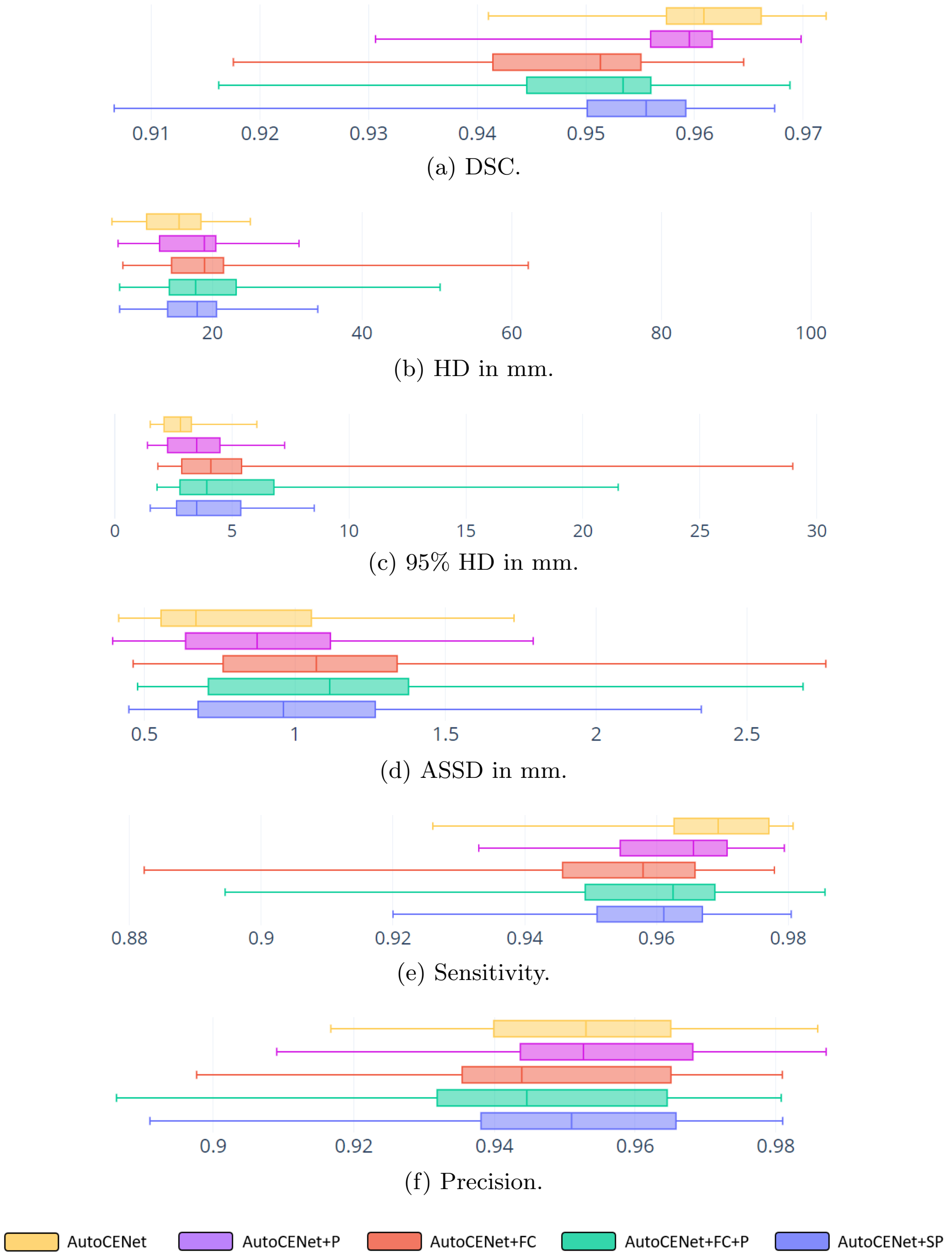
(c) 95% HD in mm.

(d) ASSD in mm.

(e) Sensitivity.

(f) Precision.

Figure 3.26: Box plots of the evaluation metrics for contour variants.



(a) DSC.

(b) HD in mm.

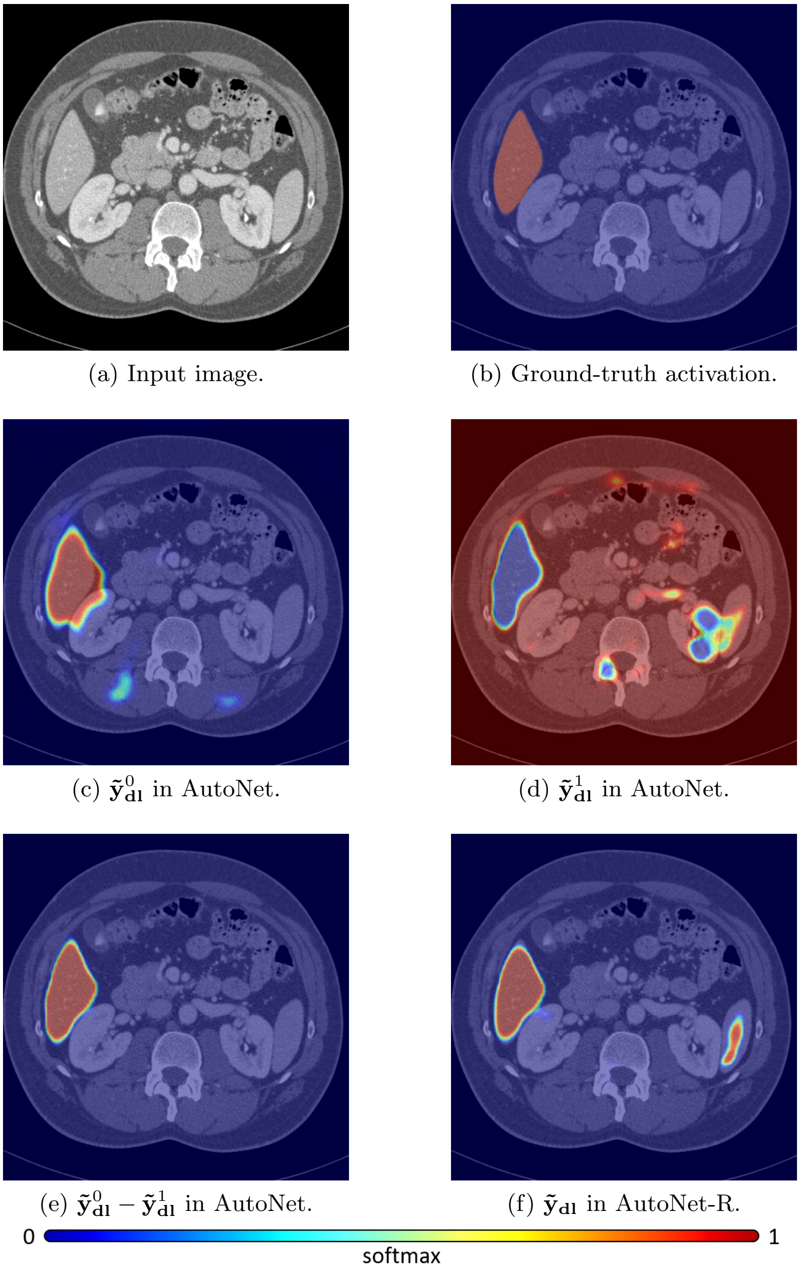
(c) 95% HD in mm.

(d) ASSD in mm.

(e) Sensitivity.

(f) Precision.

Figure 3.27: Box plots of the evaluation metrics for contour variants (CCA post-processing).



(a) Input image.

(b) Ground-truth activation.

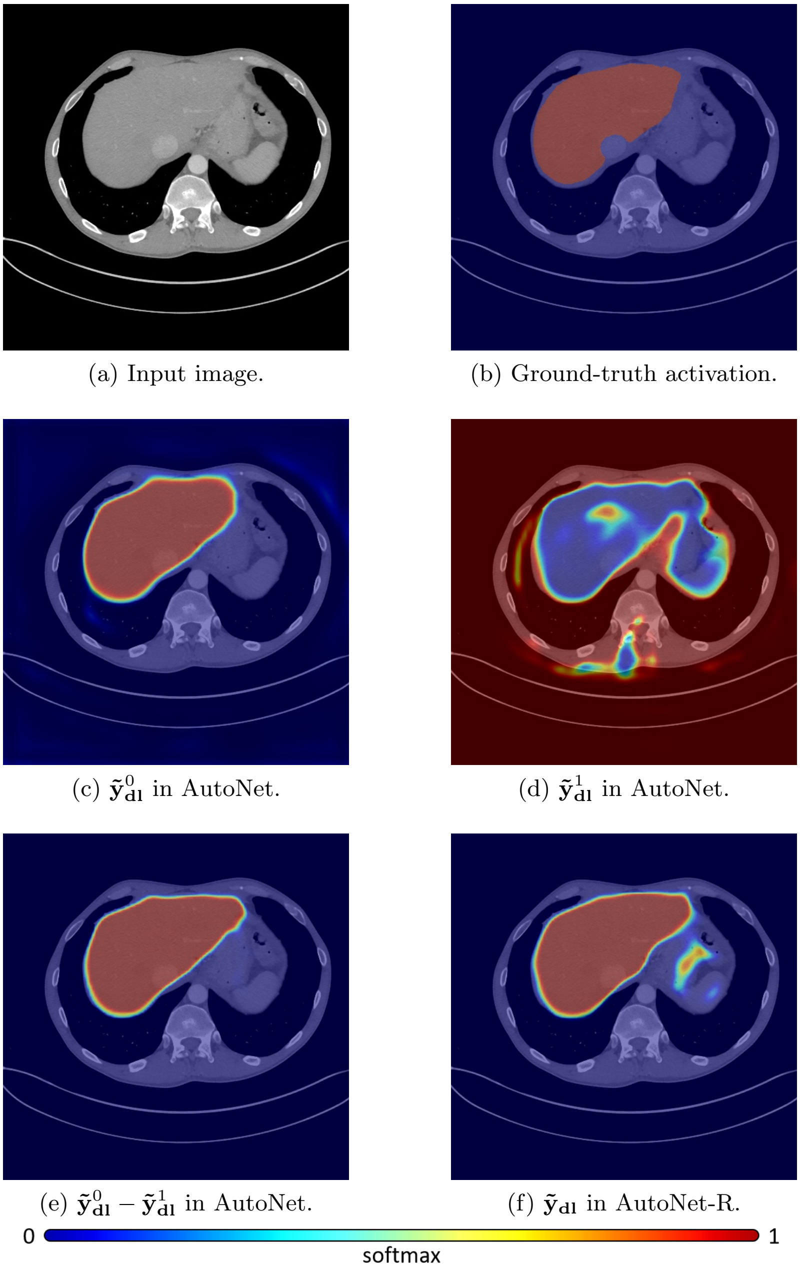
(c) $\tilde{\mathbf{y}}^{0}_{\mathbf{dl}}$ in AutoNet.

(d) $\tilde{\mathbf{y}}^{1}_{\mathbf{dl}}$ in AutoNet.

(e) $\tilde{\mathbf{y}}^{0}_{\mathbf{dl}} - \tilde{\mathbf{y}}^{1}_{\mathbf{dl}}$ in AutoNet.

(f) $\tilde{\mathbf{y}}_{\mathbf{dl}}$ in AutoNet-R.

Figure 3.28: Liver prior estimations by the AutoNet and AutoNet-R.



(a) Input image.

(b) Ground-truth activation.

(c) $\tilde{\mathbf{y}}_{\mathbf{dl}}^{0}$ in AutoNet.

(d) $\tilde{\mathbf{y}}_{\mathbf{dl}}^{1}$ in AutoNet.

(e) $\tilde{\mathbf{y}}_{\mathbf{dl}}^{0} - \tilde{\mathbf{y}}_{\mathbf{dl}}^{1}$ in AutoNet.

(f) $\tilde{\mathbf{y}}_{\mathbf{dl}}$ in AutoNet-R.

Figure 3.29: Liver prior estimations by the AutoNet and AutoNet-R.

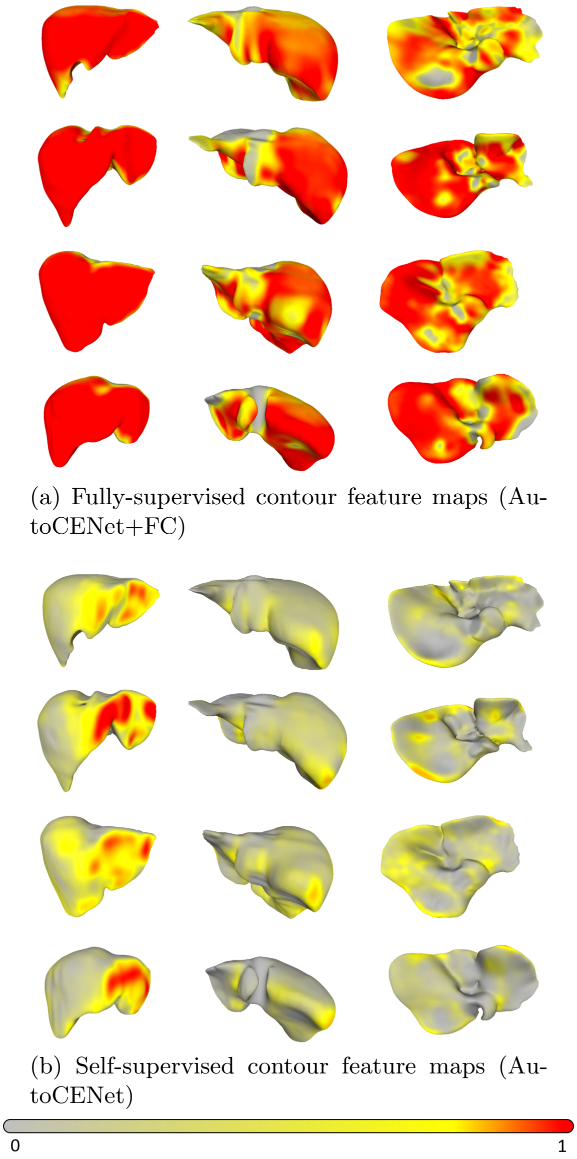
Contour feature layer

In this section, the contour features that were used to perform contour attention are studied. The fully-supervised (3.18) and self-supervised (3.9) contour feature maps are visualized in Fig. 3.30.

The contour feature map of a fully supervised network (i.e., using ground-truth contour supervision without self-supervision (3.18)) was activated within overall contour regions (Fig. 3.30a). The figure demonstrates that even with the fully supervised training, the network failed to extract full-contour features accurately (i.e., a part of the low softmax responses on the ground-truth contour region). On the other hand, with a self-supervised network, the contour feature map was activated in the local, sparse contour regions (Fig. 3.30b). The sparse contour feature map acted as attention that the network can concentrate more on the accurate delineation of boundary regions. By using the self-supervised contours, the network resulted in the improvement of the final segmentation.

The analysis of the contour learning mechanism is presented in Fig. 3.31. The figure illustrates the final output prediction of AutoNet and the following two networks: AutoCENet and AutoCENet+FC. The two networks were trained by using contour self-supervision and full-contour supervision from the fully trained AutoNet model visualized in Fig. 3.31 at first column. The colors mapped for each surface of the prediction in Fig. 3.31 represent the Euclidean surface distance to the ground-truth surface (in case of result) and a softmax value normalized into the range [0..1] (in case of contour). Each contour response is visualized based on the ground-truth surface.

As shown in Fig. 3.31, self-supervised contour responses did not correspond to the initial, weak contours from AutoNet (i.e., the initial sparse contour supervision starts from the weak parts of AutoNet results). A strong indication of


(a) Fully-supervised contour feature maps (AutoCENet+FC)

(b) Self-supervised contour feature maps (AutoCENet)

Figure 3.30: Contour feature visualizations after full training: (a) with full-contour supervision and (b) with self-supervision. The self-supervised contour feature map in (b) is sparser than that of the full-supervision and is later used as strong contour attention. The ground-truth surface is used for visualizing the distribution of the contour feature. The softmax value is normalized into the range [0..1].

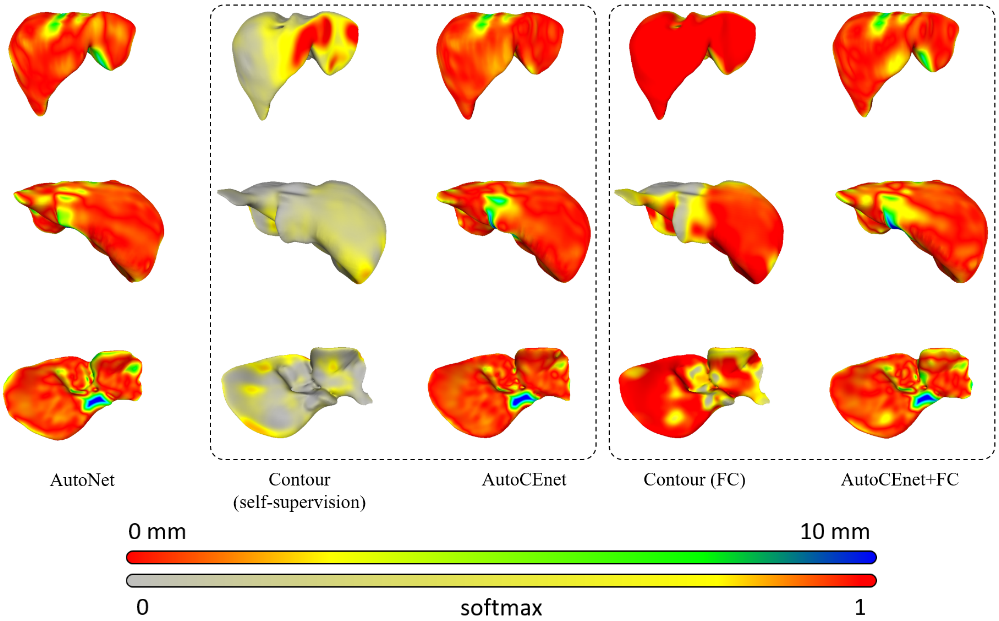
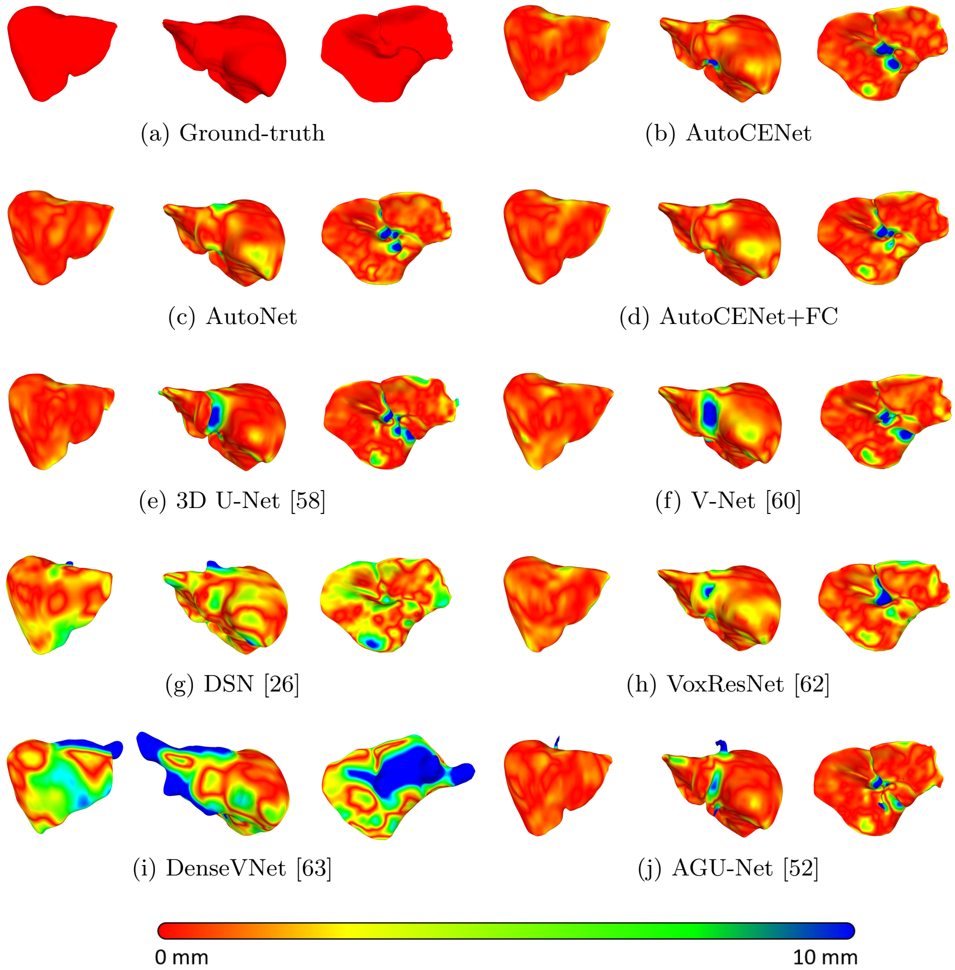


Figure 3.31: AutoNet result and contour extensions.

Fig. 3.31 is that the self-supervised contour feature guides the network to better delineate object contours rather than learning the misclassified counterparts. That is, the response of the contour feature successively changes regarding the current output prediction which acts as implicit attention for the network. Note that the contour features are not complementary that are merged for the final output. Additional visualization of the results is presented in Fig. 3.32.

3.5.6 Performance of generalization

Prior researches have thoroughly investigated neural networks in an architectural view and verified their performances within individual metrics. However, little academic research has been conducted to show the performance of generalization. To evaluate the performance of generalization, N-fold cross-validations are demonstrated for the presented networks. Figures 3.33, 3.34, and 3.35 illustrate dice loss of test images (i.e., 80 images) by training the network by



(a) Ground-truth

(b) AutoCENet

(c) AutoNet

(d) AutoCENet+FC

(e) 3D U-Net [58]

(f) V-Net [60]

(g) DSN [26]

(h) VoxResNet [62]

(i) DenseVNet [63]

(j) AGU-Net [52]

Figure 3.32: Example visualizations of the test results. The surface color is visualized based on the distance to the ground-truth surface. Visualized surfaces are smoothed by the curvature flow smoothing method [148] at the original image resolution.

using 10%, 30%, 50%, 70%, and 90% of training images out of 100 images. This experimental setting approximately proxies the real-life deep learning problem and shows an extremely generalized regularization analysis.

The overall test errors increased in a lower percentage of training images.

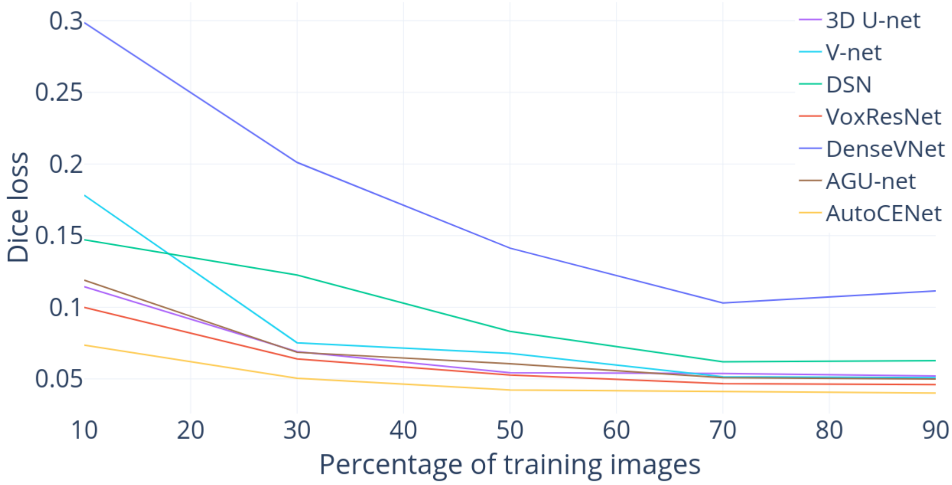


Figure 3.33: N-fold cross-validation study of AutoCENet and state-of-the-art networks. The errors are calculated based on 80 test images using dice loss.

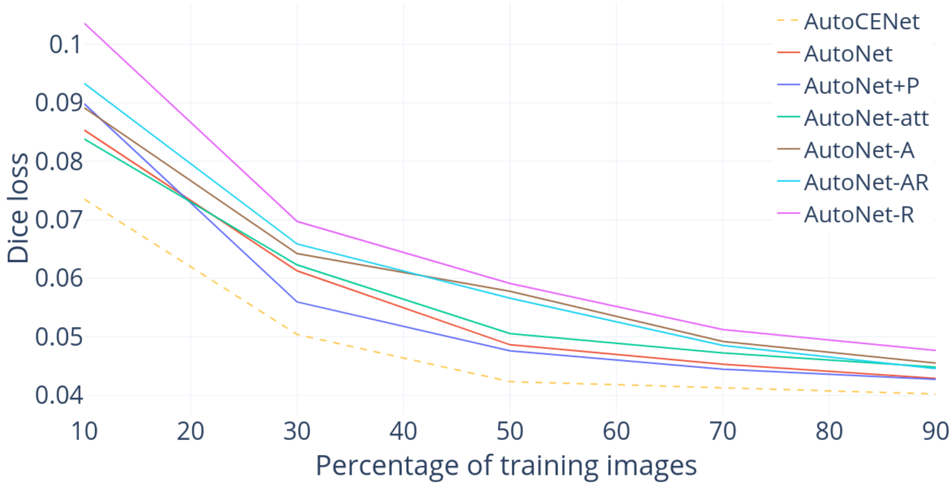


Figure 3.34: N-fold cross-validation study of AutoNet and the ablations. The errors are calculated based on 80 test images using dice loss.

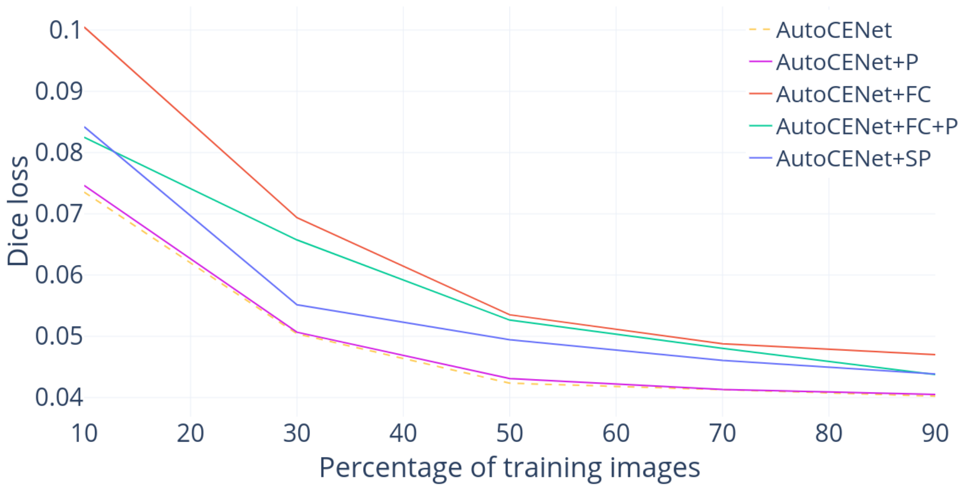


Figure 3.35: N-fold cross-validation study of AutoCENet and the contour variants. The errors are calculated based on 80 test images using dice loss.

The proposed AutoCENet showed the best performance of generalization. AutoCENet relatively did not over-fitted (i.e., lowest test error) to the training images compared to the other networks. The VoxResNet [62] was the second-best out of other state-of-the-art networks. The fair performance of VoxResNet is obtained by an auto-context algorithm [62]. Severe errors of DenseVNet [63] were caused by weak representative shape-prior as in the aforementioned evaluations.

The ablation networks of AutoCENet showed comparable performances to the other state-of-the-art networks (Figs. 3.33, 3.34, and 3.35). Among AutoNet variations, AutoNet-A, AutoNet-AR, and AutoNet-R showed worse performance. Especially, AutoNet-R was the worst indicating that residual shape-prior estimation plays an important role in an auto-context algorithm (Fig. 3.34). In cases of contour variants, full-supervision of contour (AutoCENet+FC) showed the worst performance (Fig. 3.35). The penalization of confident output

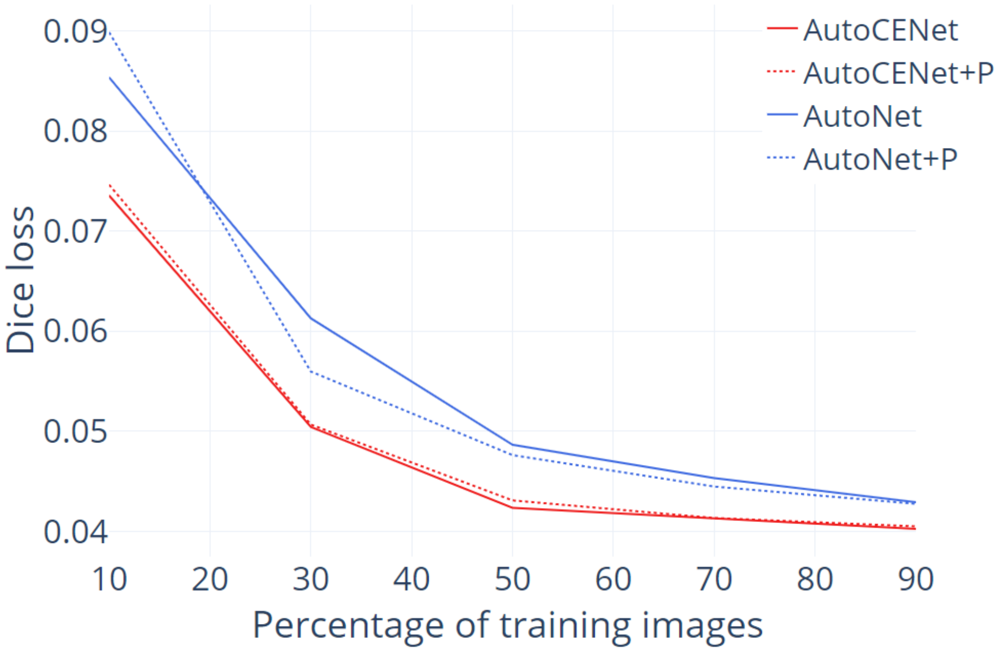


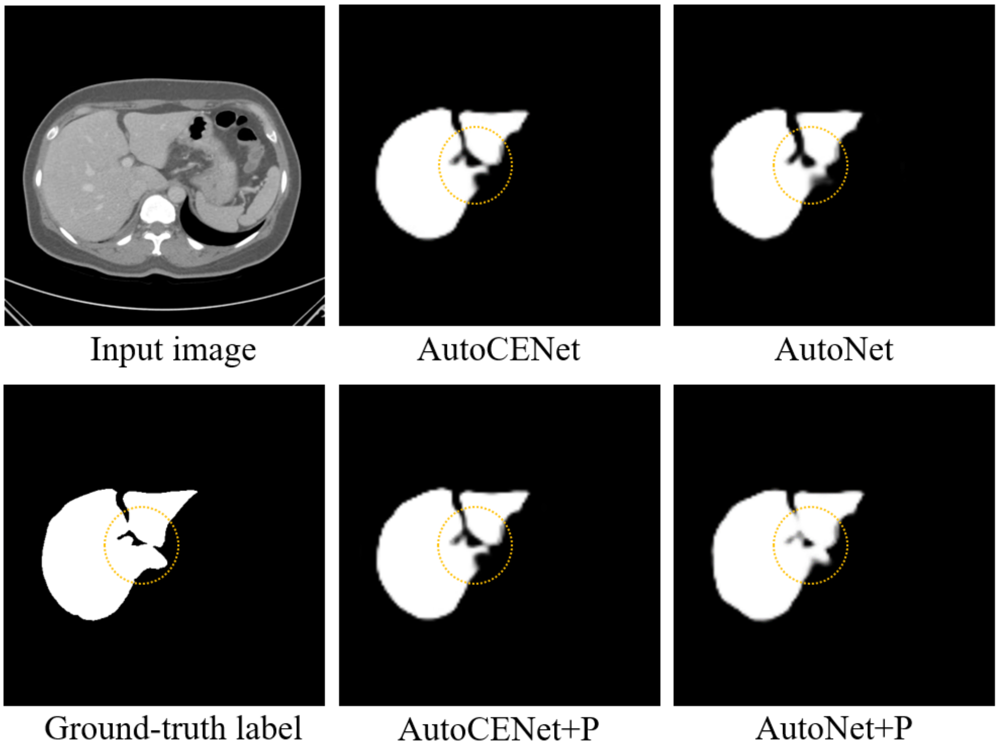
Figure 3.36: The effect of penalizing confident output loss for AutoNet and AutoCENet.

distribution improved the performances of AutoNet (i.e., AutoNet+P). However, the same penalization applied to AutoCENet showed little difference indicating that the output prediction of AutoCENet is relatively discriminative. AutoCENet+P has a negative effect, but less significant, which decreased the accuracy from the base AutoCENet.

Comparative analysis for the impact of penalizing confident output distribution is presented in Figs. 3.36 and 3.37.

3.5.7 Results from ground-truth variations

In this section, several failure cases are presented. Most of the failures are caused by the variation of ground-truth annotations. There are two notable regions: aorta and portal vein entry. As shown in Fig. 3.38, there exists severe intra-



Input image

AutoCENet

AutoNet

Ground-truth label

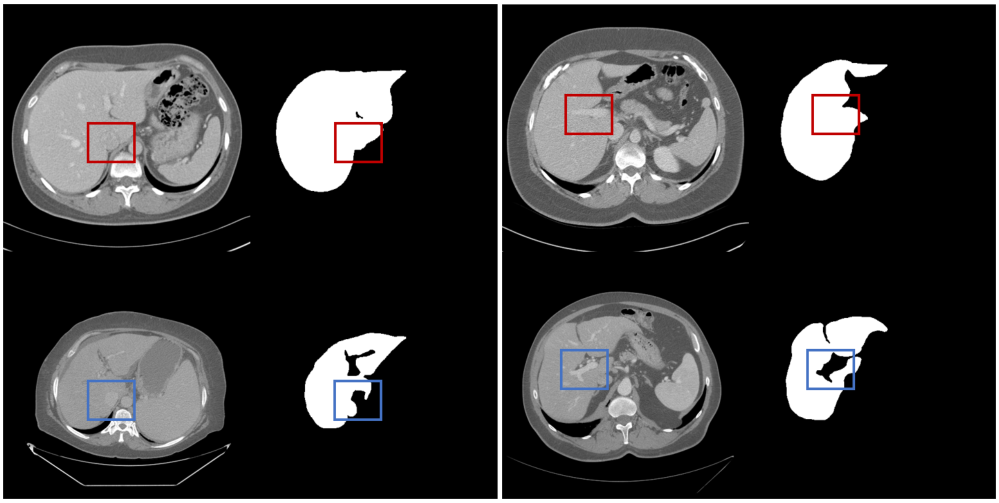
AutoCENet+P

AutoNet+P

Figure 3.37: The effect of penalizing confident output loss for AutoNet and AutoCENet.

observer variability on manual annotations. Some observers include aorta as a liver foreground, on the contrary, others exclude (3.38a). In the same sense, the region of the portal vein is included or excluded depending on the observer (Fig. 3.38b). These regions are regarded as a noise label while training. It is very hard to rely on the neural networks to automatically identify accurate annotations.

Figure 3.39 shows an example visualization of the AutoCENet result. The boundary region within the aorta and the hepatic vein is ambiguous (i.e., smoothed). The accuracy evaluation and visualizations in the previous sections were affected by this effect.



(a) Two different ground-truth annotations for aorta region.

(b) Two different ground-truth annotations for portal vein region.

Figure 3.38: Variations on ground-truth segmentation labels. (a) Aorta region and (b) portal vein entry region.

3.6 Discussion

In recent years, the employment of shape priors or neural networks has been the most promising method for the accurate segmentation of a liver. The proposed network avoided using the shape priors because the performance can be seriously dependent on the variability of the shape priors. If the training set is insufficient, the algorithm breaks down owing to the learned prior. The proposed auto-context neural network internally used a high-level residual estimation of shape-prior to robustly acquire the posterior probability. The embedded liver probability map acts as a post-inference prior, which can be further used for the final accurate segmentation. As a result, a single-passing auto-context neural network was established without separate classification series as in [11, 62, 73]. The main underlying principle of the base auto-context architecture is that the

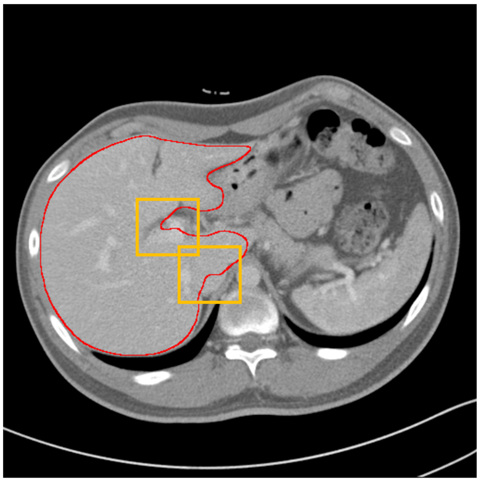


Figure 3.39: An example result of AutoCENet. Boundaries within the aorta and hepatic vein regions were smoothed.

performance of generalization can be achieved by a robust estimation of the overall shape of a liver. In that perspective, high-level residual shape estimation in a lower resolution successfully attained the desired task without extra parameters of complex neural structures.

The attention mechanism has been growing its applicability to be a dominant method for modern neural networks. However, the self-attention mechanism still has a limitation on the fact that it is a data-driven algorithm, paradoxically. The method cannot outperform without explicit general guidance on certain applications especially that suffers from data deficiency. In the application of liver segmentation, it is very hard to make a neural network to pay attention to certain features that are useful for improving the final output. In other words, complementary feature learning (i.e., special penalization presented in the experiments) has failed, and there is no existing method to make the network to attend the failures. In this dissertation, a self-supervised contour delineation is applied to the intermediate feature that is intended to implicitly guide the network to give more attention to weak boundary regions that the network has failed to accurately delineate. The self-supervising mechanism successfully embedded in the network and improved the final accuracy. The overall architecture of the proposed neural network, which exploited an auto-context and the contour self-supervision, suggests that performance of generalization and accuracy can be obtained by a human-designed curriculum which means a domain-specific knowledge is still required in the modern application of neural networks.

Chapter 4

Liver Vessel Segmentation via Active Contour Model with Dense Vessel Candidates

4.1 Overview

Accurate vascular structure analysis of a liver is an essential procedure of clinical diagnosis. For example, anatomical liver partitioning can be obtained by accurate analysis of portal vein structure. Thus, accurate segmentation of vessels can bring an accurate structural analysis of the liver, and eventually, improve clinical studies regarding diagnosis and surgical planning (e.g., liver resections). However, it is very difficult to depict a complex 3D model like liver vessels even for the highly educated experts in the clinics. In this chapter, a method of automatic segmentation of liver vessels is presented. The method employs a liver prior (i.e., segmentation results) that was obtained by a liver segmentation proposed in the previous chapter.

To tackle the challenge of vessel segmentation, a fully automated and new

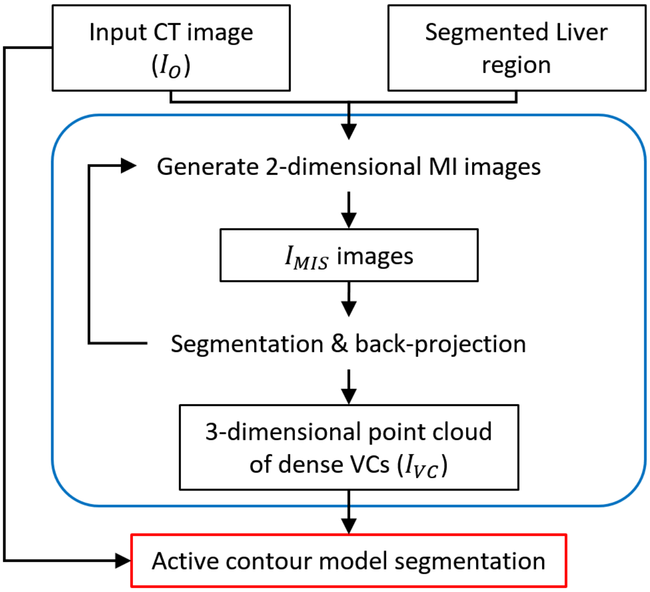

Input CT image (I_O)
Segmented Liver region
Generate 2-dimensional MI images
I_{MIS} images
Segmentation & back-projection
3-dimensional point cloud of dense VCs (I_{VC})
Active contour model segmentation

Figure 4.1: Overall workflow of the proposed method. The blue area illustrates the iterative method of maximum intensity slab image generation, segmentation, and back-projection. The proposed active contour model based on the level set is performed after I_{VC} generation (red).

liver vessel segmentation is proposed which includes portal and hepatic veins (i.e., all contrast-enhanced vessels). A novel dense vessel candidate points are introduced to enhance the robustness and a new level set functional is proposed (Fig. 4.1). First, similar to sliding-thin-slab image analysis [150], the 3D segmentation problem is reduced to a 2D problem by generating maximum intensity images within the slabbed region rather than the full projection of the original volume (Fig. 4.2). The reason for not using a full projection of maximum intensity image is that it is very difficult to segment detailed vessels, and it is hard to back-track the original position in the 3D domain due to extreme overlaps (Figs. 4.3, 4.4, and 4.5). A new 2D vessel segmentation algorithm is then performed and the result pixels are back-projected to the original 3D space

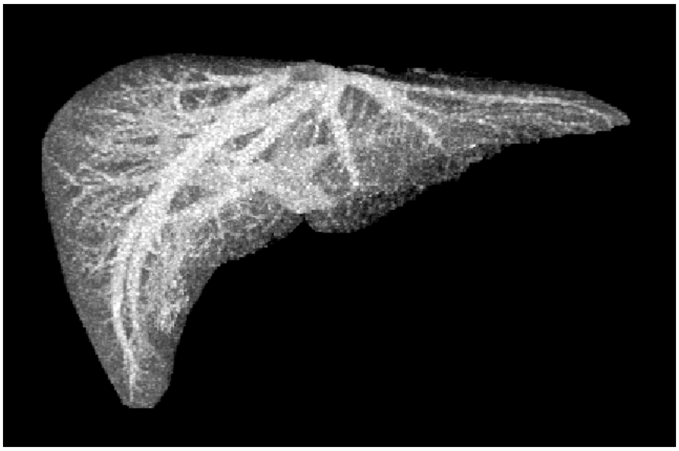


Figure 4.2: Maximum intensity projection image for y-axis direction. The input of the segmented liver is used to project intensities of voxels that are inside the liver region.

to generate vessel candidate points. The map of very dense vessel candidates is generated by multiple maximum intensity images. From point cloud generated by dense vessel candidates, the accurate segmentation of a vessel region in 3D is performed by a newly designed active contour model. This is a very hard problem because of the three challenging requirements: 1) making an accurate and smooth boundary of an object, ignoring holes in vessel candidate point cloud if exist, and fine-tuning vessel region with the original 3D image. The vessel candidate cloud is dense enough to define a region but there is a lot of empty spaces and noisy points in the boundary area. Many probability density estimation methods are mostly parameter-dependent and difficult to find accurate object boundaries even with non-parametric estimations. To resolve the aforementioned issues, a new active contour model is proposed for the accurate segmentation of vessel structures.

As illustrated in Figs. 4.2 and 4.5, maximum intensity projection without introducing slab is infeasible for vascular structure analysis. Although a domain

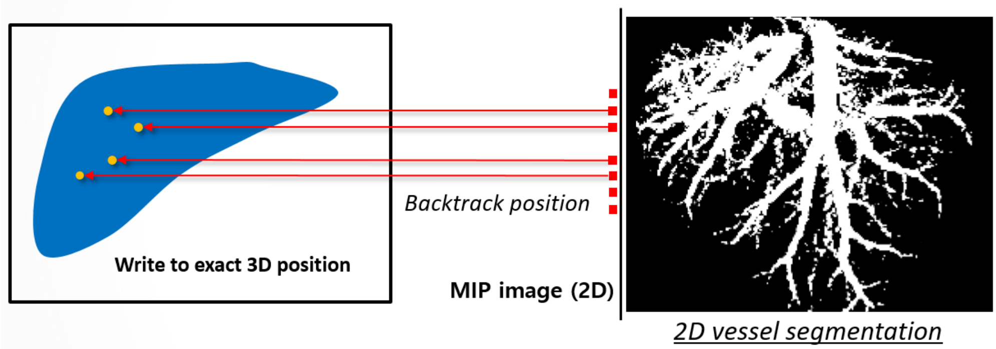


Figure 4.3: Segmentation of vessel on 2D maximum intensity projection (MIP) image and back-projection. The MIP image represents a single maximum intensity-valued position for each pixel. It is difficult to reconstruct 3D positions of vessels in terms of structural analysis. The extremely complex vascular structure is highly overlapped in a single MIP image.

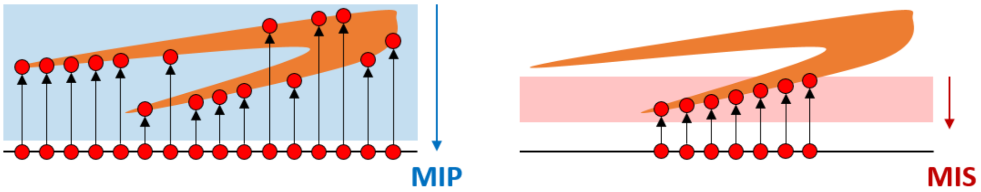


Figure 4.4: An example backtracking of maximum intensity projection (MIP) and maximum intensity slab (MIS) images.

reduction from 3D to 2D can ease the task of segmentation, extreme overlaps of vessels make the image noisy that structural analysis becomes challenging. Several methods have been proposed to tackle the task of 3D vessel segmentation by employing maximum intensity projections [74–76]. A common approach is to perform segmentation on a 2D projection image to obtain seed points for further 3D segmentation such as region-growing or thresholding [75, 76]. However, performing segmentation on maximum intensity projection images is challenging (Fig. 4.5). A simple threshold-based segmentation, directly applied to the image, result in noisy foregrounds that are unreliable (Fig. 4.5). Furthermore,

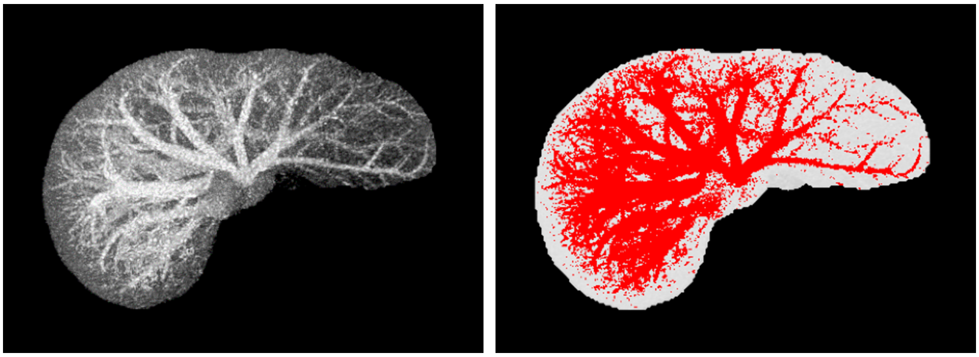

(a) (b)

Figure 4.5: An example maximum intensity projection (MIP) image with respect to z-axis and a manual threshold result. It is difficult to segment accurate vessels from MIP image. Many noise and false positives are made by thresholding.

depth values in depth-buffer, presented in [76], can be inconsistent regarding single vascular structure due to highly overlapping vessels. The region-growing method for the final 3D segmentation is also vulnerable because of the high noise variance in the original image. In this dissertation, a robust 2D segmentation method is proposed which employs the strength of a maximum intensity projection and avoids the weakness of a full 3D projection.

The proposed method has two major advantages. The one is a problem domain reduction from 3D to 2D by a maximum intensity projection scheme. This domain reduction makes segmentation tasks simple and robust because of the degraded dimensionality and the lowered noise variation. To reduce the complexity of structural analysis regarding vascular structure in projected images, a slab-based projection is proposed. By employing slab-based maximum intensity projection, clear vascular structures such as lines and blobs can be more salient. The eased 2D image segmentation process contributes to the generation procedure of accurate dense vessel candidate points, especially in a weak vessel

Algorithm 1: Selecting Three Optimal Clusters.

```
Input: $I_o$ (input CT image), $L$ (segmented liver labeled region).
Output: Segmented vessel labeled region.
1. Dense vessel candidate generation:
   a) Generate maximum intensity slab image ($I_{MIS}$) within $L$
      (i.e., domain reduction, $3D \longrightarrow 2D$).
   b) Apply BM3D denoising to $I_{MIS}$.
   c) Apply vessel enhancement filtering to $I_{MIS}$.
   d) Apply threshold to $I_{MIS}$ to acquire vessel region foreground.
   e) Back-project foreground pixels of $I_{MIS}$ to $I_{VC}$ vessel
      (i.e., domain restoration, $2D \longrightarrow 3D$).
   f) Iterate over a)-e) for the three axis-aligned sliding slabs.
2. Clustering of dense vessel candidates:
   a) Smooth binary image: $I_{VC} \longrightarrow I_{GVC}$.
   b) Perform active contour model segmentation with $I_o$ and $I_{GVC}$.
```

region. The second is the application of dense vessel candidate points to the new level set-based active contour model. Dense vessel candidate points act like high probability seed points that guide an active contour model to extract a more accurate vessel region in the original 3D image. With the help of dense vessel candidate points, the method can easily extract thin and weak peripheral branch vessel structures whose boundaries are hard to identify in the original CT images. Moreover, the difficulty of an initial contour setting in a level set method is resolved by estimating the initial contour based on the generated vessel candidate image. That is, the generated vessel candidate map (i.e., binary) is dilated which serves as an initial estimation of the vessel contour. An overall procedure of the presented algorithm is presented in Algorithm 1.

4.2 Dense vessel candidates

In this section, a generation process of 3D binary map formed by dense vessel candidates is proposed. A maximum intensity projection is performed based

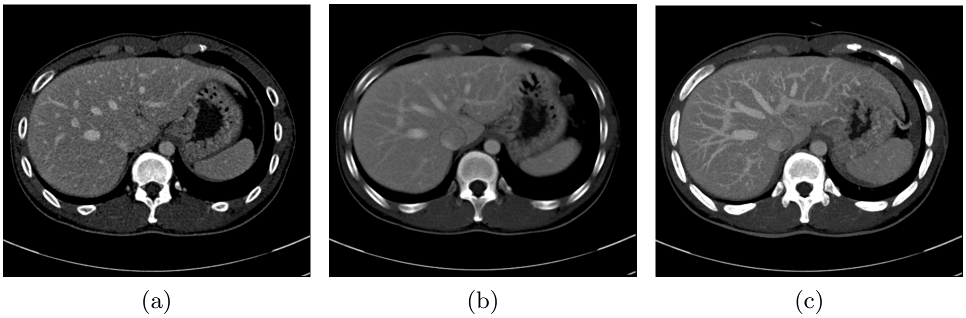
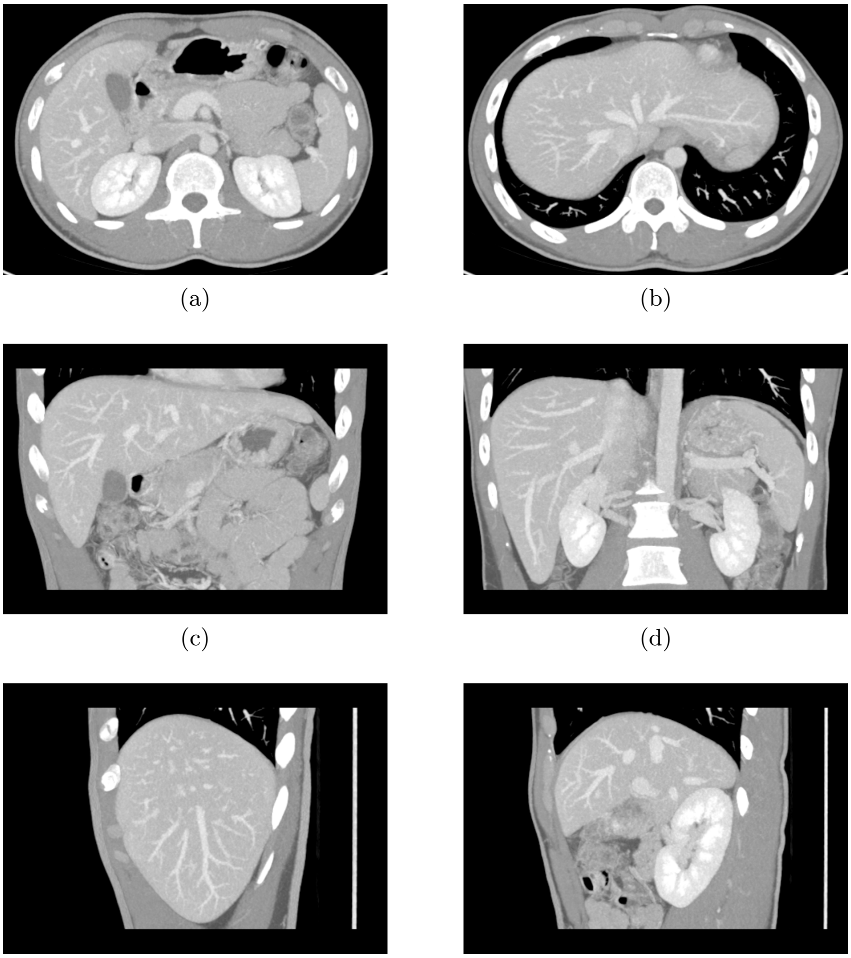


Figure 4.6: Axial image comparison between (a) single-thickness plane image, (b) slab-thickness plane image with an averaging scheme, and (c) slab-thickness plane image with maximum intensity scheme (I_{MIS}). Maximum intensity-based projection (c) represents better (i.e., clear and salient) vessel structures without noise compared to that of averaging (b).

on slabs that are a plane-based volume which is constrained by a given thickness. A sequential procedure of projection, segmentation, and back-projection is performed for each sliding slabs. For a robust segmentation on 2D projection images, additional denoising and vessel enhancement filtering are applied before segmentation.

4.2.1 Maximum intensity slab images

To generate vessel candidates, maximum intensity-based planar reconstruction is performed based on the slab region rather than full projection (Fig. 4.2). Defining the original image as $I_o : (x, y, z) \longrightarrow \mathbb{R}$ and the maximum intensity slab (MIS) image as $I_{MIS} : (x, y) \longrightarrow \mathbb{R}$, the reconstruction process can be represented by $I_o \longrightarrow I_{MIS}$ which is a domain reduction from 3D to 2D. The maximum intensity slab image is produced by storing only the maximum intensity value from the 3D data encountered by a plane's normal directional ray casting through an object. By the dimensions of an input image w (width), h (height), and d (depth), a vector in an image coordinate can be defined as



(a)

(b)

(c)

(d)

(e)

(f)

Figure 4.7: Maximum intensity slab images based on multiple axis-aligned reconstructions. (a-b) Axial, (c-d) coronal, (e-f) sagittal views, respectively.

$\mathbf{v_i} = (x, y, z)$, where $0 \le x \le w$, $0 \le y \le h$, and $0 \le z \le d$. Introducing a normal vector, $\vec{n}$, of a plane, MIS image can be represented as

$$I_{MIS}(\mathcal{P}(\mathbf{v_p^{\tilde{n}}})) = \max_{-\sigma/2 \le p \le \sigma/2} I_o(\mathbf{v_p^{\tilde{n}}} + p \cdot \vec{n}), \tag{4.1}$$

where $\mathcal{P}$ is a domain projection from 3D to 2D (i.e., coordinate transformation), $\mathbf{v_p}\tilde{\mathbf{n}}$ is a projected plane coordinates on 3D based on normal vector $\vec{n}$, and p is

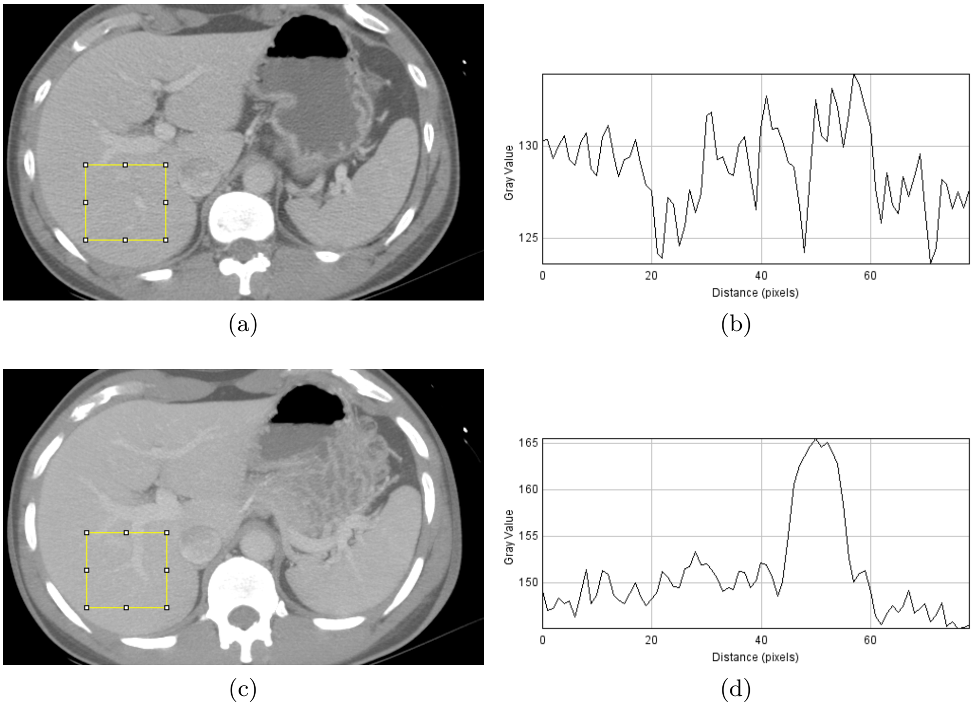
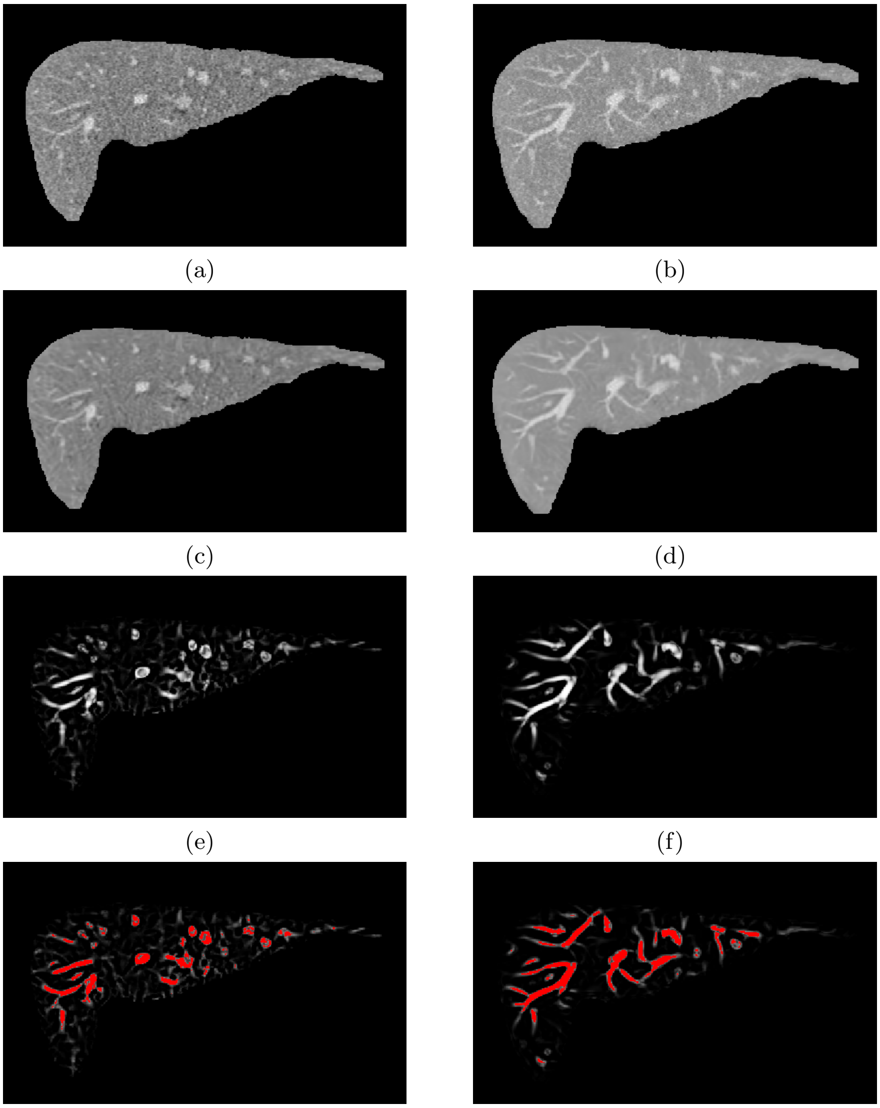


Figure 4.8: A simple planar reconstructed image (the first row) and maximum intensity projected image within the slab (the second row). (a) A 1-voxel thickness reconstructed an axial image. (b) The intensity profile of a given rectangle in an image (a). (c) A 7-voxel thickness reconstructed an axial image at the same position as (a). (d) The intensity profile of a given rectangle in an image (c). The intensity profile shows that multiple image projection (i.e., maximum projection) results in noise reduction regarding the background and foreground vessels.

an offset value from the projection plane. The slab is defined by a thickness parameter, σ. Depth information is also stored for further back-projection. The selection of the maximum intensity value decreases the variance of the background values that appear in the maximum intensity projection image [76]. Boundaries of vessels including minor ones are thus far more salient than those in the original image (Fig. 4.6). Furthermore, more information on vessel structure can be contained than a single-depth planar reconstructed 2D image. Slabbed planar reconstruction by averaging scheme also cannot fully represent fine details of minor vessels (Fig. 4.6b). One of the main ideas of the proposed method is to make 2D segmentation more robust by using maximum intensity slab image (Fig. 4.6c) rather than using a simple 2D plane image (Fig. 4.6a) to get more accurate and dense vessel candidates. The major advantages of maximum intensity slab image are that it is a naturally denoised image with maximum intensity projection scheme and the vessel line structure is more delineated for further responses of vessel enhancement filter. The background noises are suppressed so that the difficulty of the 2D vessel segmentation problem can be relaxed (Fig. 4.8). As shown in the first column of Fig. 4.9, with a single-depth planar reconstructed image, vessel candidates cannot be extracted accurately even with extra denoising and vessel enhancement techniques.

In the proposed method, voxels only in the liver region are calculated using liver segment input (Fig. 4.9). In this way, the maximum intensity slab image is not affected by the intensity distribution of the outer liver region. For fine details, we reconstruct maximum intensity planar images for multiple directions of the plane normal (4.1). For each axis-aligned direction (e.g., $\vec{n} = (1, 0, 0)$ for x-axis direction), slabbed plane is shifted for a single voxel distance (Fig. 4.11). The maximum intensity slab image is defined by a plane centered at slab region (schematic illustration; the formal representation is defined in (4.1)). A shift of a maximum intensity slab image is very important because by performing maximum intensity slab shifting, very dense vessel candidates can be later obtained that can help to segment the vessel region in a 3D image using an active contour model.



(a)

(b)

(c)

(d)

(e)

(f)

(g)

(h)

Figure 4.9: The comparison of 2D segmentation results based on a simple plane image (i.e., 1-voxel thickness; first column) and maximum intensity slab (MIS) image, I_{MIS} (second column) with respect to y-axis direction: (a) Simple plane image without thickness; (b) MIS image from I_o with 7 slab thickness. Each following row applies BM3D denoising [151], vesselness filter [102], and thresholding to the previous row image, respectively.

4.2.2 Segmentation of 2D vessel candidates and back-projection

Once MIS images are generated, block matching 3D (BM3D) denoising technique [151] and multi-scale vessel enhancement filtering [102] are sequentially performed to I_{MIS} images (Fig. 4.9).

Block matching 3D (BM3D)

A huge body of literature has been studied for image denoising from simple smooth filtering to more complex approaches: non-linear filterings [152, 153], partial differential equations [154, 155], non-local statistics [156–159], and transform-domain filtering [151, 160, 161]. Some approaches assumes that a true signal (denoised) and noise can be separated via variational modeling [154, 155]. Non-local means method [156] suggests an averaging scheme with non-local patches. The transform-domain denoising technique [151, 161] attempts to model true signal in a frequency domain that can be recovered to the original images.

Among the literature of image denoising, block matching 3D (BM3D) [151] is employed in the proposed workflow. The method employs non-local image patches as proposed in [159]. The basic idea of a non-local means method is to build a point-wise estimate of the image where each pixel is obtained as a weighted average of pixels centered at regions that are similar to the region centered at the estimated pixel [151]. The authors in [151] proposed transform-domain sparse filtering with an adaptation of the non-local method. The enhancement of sparsity is achieved by grouping similar 2D fragments of the image into 3D data arrays, and subsequently, perform collaborative filtering to 3D stacked data [151]. The collaborative filtering reveals even the finest details shared by grouped fragments and at the same time, it preserves the essential unique features of each fragment [151].

Multi-scale vessel enhancement filtering

A multi-scaled vessel enhancement filtering has been proposed in [102]. Second-order analysis of local feature of an image, Taylor expansion in the neighborhood of a point x can be defined as

$$L(x + \delta x, s) \approx L(x, s) + \delta x^T \nabla_s + \delta x^T \mathcal{H}_s \delta x, \tag{4.2}$$

where ∇_s and $\mathcal{H}_s$ are the gradient vector and Hessian matrix of the image computed in x at scale s [102]. By employing a linear scale space theory [162, 163], scaled differential operator of L can be defined as a convolution with derivatives of Gaussians [102]:

$$\frac{\partial}{\partial x} L(x, s) = L(x) * \frac{\partial}{\partial x} G(x, s) \tag{4.3}$$

where the scale s is defined as a standard deviation of the Gaussian function G_s. The Hessian matrix is similarly defined as:

$$\mathcal{H}_s = \frac{\partial^2}{\partial x^2} I(x, s) = I(x) * \frac{\partial^2}{\partial x^2} G(x, s), \tag{4.4}$$

where I is a given image (Fig. 4.10).

The eigenvalue analysis of the Hessian matrix is to extract the principal directions where the local second-order structure of the image can be decomposed [102]. Defining eigenvalues of the Hessian as $|\lambda_1| \leq |\lambda_2| \leq |\lambda_3|$, several structural analysis can be performed by using the values. For example, an ideal tubular structure in a 3D image can be:

$$|\lambda_1| \approx 0, \tag{4.5}$$

$$|\lambda_1| \ll |\lambda_2|, \tag{4.6}$$

and

$$\lambda_2 \approx \lambda_3. \tag{4.7}$$

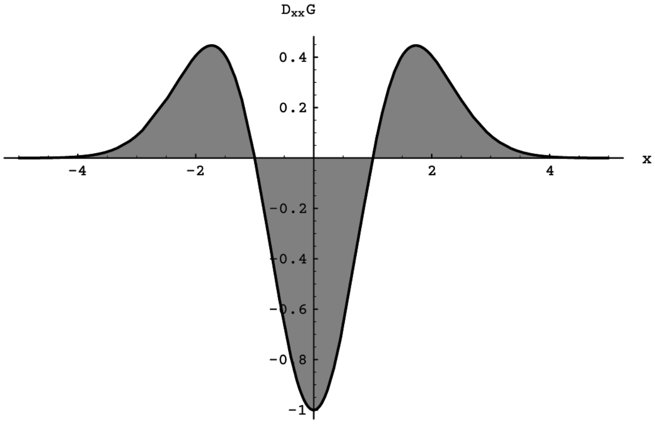


Figure 4.10: The second order derivative of the Gaussian kernel ($s = 1$) [102].

The intuition comes from the fact that a pixel belonging to a vessel region will be signaled by λ_1 (i.e., minimum principal direction) being small, and λ_2 and λ_3 of a large magnitude ane equal sign [102].

In the proposed vessel enhancement in a 2D domain, the blobness measure and contrast measure compared to the background are used. The blobness measure accounting for the eccentricity of the second-order ellipse is defined as

$$\mathcal{R}_B = \frac{\lambda_1}{\lambda_2}, \tag{4.8}$$

where $|\lambda_1| \leq |\lambda_2|$. For contrast measure compared to the background, Frobenius matrix norm is used since it has a simple expression in terms of the eigenvalues when the matrix is real and symmetric [102]:

$$\mathcal{S} = ||\mathcal{H}||_F = \sqrt{\sum_{i \leq 2} {\lambda_i}^2}. \tag{4.9}$$

The contrast measures $\mathcal{S}$ filter out low values in the background where no structure is presented and the eigenvalues are small [102]. In regions with high contrast, the norm becomes larger since either eigenvalue will be large [102].

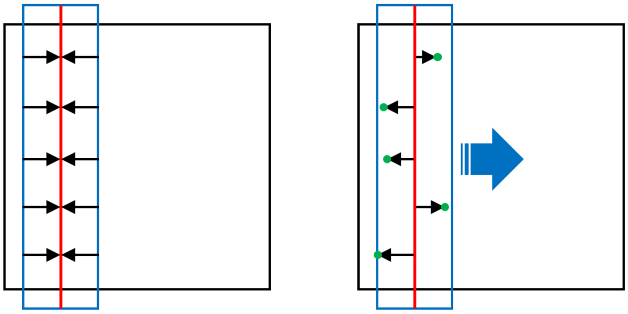


Figure 4.11: Maximum intensity slab (MIS) image (red lines) generation and back-projection. Images illustrated by black squares are 3D volumes and red lines are 2D images. Images are described in 2D for simplicity. The left image illustrates the MIS image (I_{MIS}) generation. MIS is reconstructed within a slabbed region with respect to a projection vector. The right image illustrates the back-projection mechanism. The vessel region is back-projected to the original 3D positions (green dots) to generate vessel candidates. A single voxel-sized shift interval is used and the three axis-aligned projection vectors are employed in the experiments.

The final multi-scale vessel enhancement filtering function can be defined as

$$V(I) = \begin{cases} 0, & \text{if } \lambda_2 > 0, \\ \exp(-\frac{{\mathcal{R}_B}^2}{2\beta^2})(1 - \exp(-\frac{\mathcal{S}^2}{2c^2})), & \text{otherwise,} \end{cases} \tag{4.10}$$

where β and c are thresholds that control the sensitivity of the line filter to the measures $\mathcal{R}_B$ and $\mathcal{S}$.

2D MIS segmentation

After MIS projection, BM3D denoising, and vessel enhancement filtering, vessel candidate points are finally segmented in 2D MIS images and back-projected to the original 3D domain. Writing denoised images via BM3D as I^*_{MIS}, vesselness filtering operation can be written as:

$$V(I^*_{MIS}) = \max_{1 \leq s \leq 3} V(s, I^*_{MIS}) = \max_{1 \leq s \leq 3} V(G(s) * I^*_{MIS}) \tag{4.11}$$

where $G(s)$ is a Gaussian kernel. A multi-scale analysis is performed via convolving different scales of Gaussian kernels and computing vesselness filter response by (4.11). Finally, segmentation of vessel region is performed by fix-valued thresholding (Fig. 4.9) with value 0.4, which indicates selecting a pixel with vesselness filter response greater or equal to 0.4 (Fig. 4.9). Once vessel region in multiple I_{MIS}s are obtained, foreground pixels are back-projected to 3D space defined as

$$I_{VC}(x,y,z) = \begin{cases} 0, & \text{if background} \\ 1, & \text{otherwise (vessel candidate voxel).} \end{cases} \tag{4.12}$$

I_{VC} is a 3D binary image that contains value 1 (foreground) if the corresponding voxel at that position in I_o was once or more defined as vessel region in I_{MIS} images and 0 (background) otherwise. Each foreground voxel forms vessel candidate with high probability (Fig. 4.13a). For the employment of I_{VC} on a level set functional, the binary map can be Gaussian smoothed to introduce continuous function for further gradient calculation (i.e., I_{GVC}).

The lower threshold value or automatic value (e.g., Otsu's method [164]) can be applied to extract a more accurate vessel area from a single I_{MIS}, however, this can lead to over-segmentation that generate noise in the vessel candidate set. Rough segmentation of multiple directional sliding of I_{MIS} can obtain accurate and dense vessel candidates without precise segmentation of each maximum intensity slab image. This is because vesselness filter response in I_{MIS} varies from directions. In other words, even if some vessel region points in a single I_{MIS} were lost, those points can be obtained in different directional images. By integrating all 2D segmentation results into 3D vessel candidates, noiseless and dense vessel candidates can be constructed.

4.3 Clustering of dense vessel candidates

In this section, a level set formulation of a new active contour model is presented. First, a level set formulation of the active contour model is briefly reviewed. Subsequently, the proposed energy functionals will be presented in the following subsections.

Level set formulation of active contour

An active contour model [122] based on explicitly parameterized curves can be defined in an image domain (i.e., grid) via level set [139]. Implicit representation of a curve is obtained by a zero level set function:

$$\phi(C(t), t) = 0, \tag{4.13}$$

where $C(t)$ represents a curve on level set function on time series given at time t. A level set function typically forms a signed distance form which makes contour propagation stable (Fig. 4.12). A level set speed function (i.e., contour propagation functional) can be obtained by a partial derivative regarding time t:

$$\phi_t + \nabla\phi(C(t), t) \cdot C'(t) = 0. \tag{4.14}$$

By introducing speed energy based on a normal direction, $n = \frac{\nabla\phi}{|\nabla\phi|}$, a level set formulation of an active contour can be represented as

$$\phi_t = |\nabla\phi| F, \tag{4.15}$$

where F is a speed function in a normal direction of the curve. The level set energy functional does not require the function ϕ to be a distance function, however, a typical case of implicit representation of a level set function is the signed distance function due to stability which has the property of $|\nabla\phi| = 1$

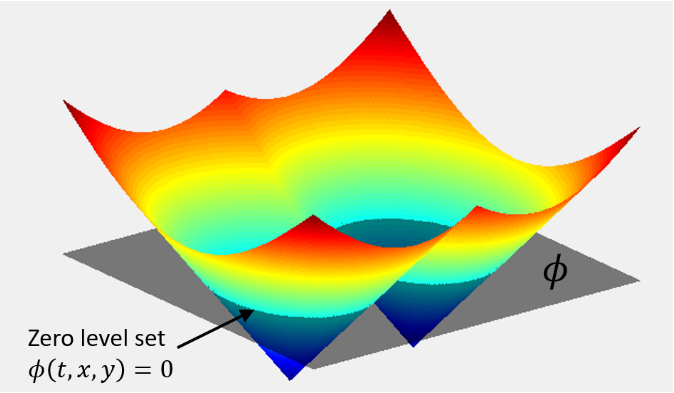


Figure 4.12: Level set function. A zero level set is defined by $\phi(t, x, y) = 0$. The function is typically formed by a signed distance function as illustrated in the figure.

that indicates the major challenging part is to design an F function for the task-dependent tasks.

4.3.1 Virtual gradient-assisted regional ACM

Regional energy term

Unlike gradient, regional information like in the Chen-Vese model [131] can be still applied to the original image, I_o. This is because the intensity distribution differs between background soft tissue and minor vessel region. However, solely using I_o can lead to inaccurate results because it is hard to globally optimize the estimation of the intensity distribution. Introducing complementary region terms by using both I_o and I_{GVC} images, a robust regional energy term can be obtained that derives a curve to converge accurately. The surface evolution in 3D space is referred to as "contour" or "curve" in the context of the dissertation for simplicity. Combining all the above features, a new virtual gradient assisted regional active contour model can be modeled.

Regional energy terms for a level set functional are presented regarding the intensity distributions. Similar to the CV model [131], the original intensity distribution of a given input image is first employed:

$$\int_{\phi} |I_G(x) - c_1|^2 H(\phi(x))dx \tag{4.16}$$

and

$$\int_{\phi} |I_G(x) - c_2|^2 (1 - H(\phi(x)))dx \tag{4.17}$$

where I_G is a Gaussian smoothed image from the original image. c_1 and c_2 are the curves's internal and external averages of I_G as similarly defined in (2.48) and (2.49). To compensate for the possibility of false dense vessel candidates, it is important to design an energy function based on the original image.

In addition to the original image, a smoothed vessel candidate map is also employed for a regional energy terms:

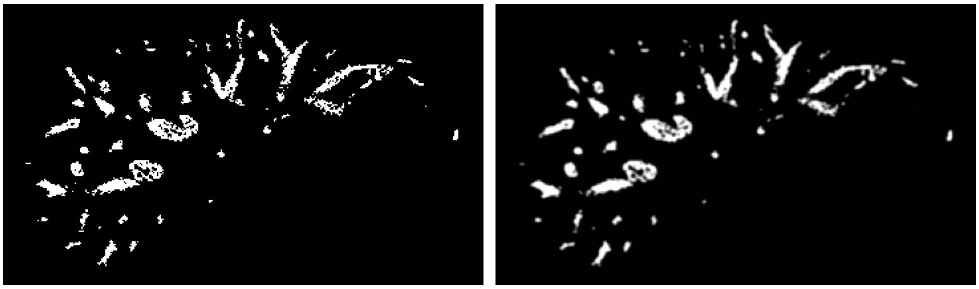
$$\int_{\phi} |I_{GVC} - g_1|^2 H(\phi(x))dx \tag{4.18}$$

and

$$\int_{\phi} |I_{GVC} - g_2|^2 (1 - H(\phi(x)))dx \tag{4.19}$$

where g_1, g_2 are regional intensity averages similarly defined as c_1, c_2, and I_{GVC} is Gaussian smoothed image from I_{VC}. The formal expression will be presented in the following sub-sections.

The major intention of introducing the two regional terms is that the two different intensity domain (i.e., the original image and smoothed vessel candidate map) can be mutually complementary. That is, a weak representation of peripheral vessels on the original image can be complemented by the vessel candidate map. Reversely, the sparse representation of thick vessels on the vessel candidate map can be complemented by the original image.



(a) (b)

Figure 4.13: (a) Dense vessel candidate image generated by segmentation and back-projection of maximum intensity slab images (I_{VC}). (b) Gaussian smoothed I_{VC} image (I_{GVC}) with standard deviation, $\sigma = 1$.

Virtual gradient energy term

Gradient stopping criteria is an important term for an object delineation via active contour models. However, it is difficult to model the gradient criterion based on the original image due to a weak representation of edges on peripheral vessels. In that perspective, a new virtual gradient is proposed to model the edge criterion for an active contour. To make a continuous function of I_{VC}, the Gaussian smoothing is firstly applied to the binary map, I_{VC}:

$$\nabla^* I_{VC} = \nabla I_{GVC} = \nabla(G_\sigma * I_{VC}), I_{GVC} : (x, y, z) \longrightarrow \mathbb{R} \tag{4.20}$$

where G_σ is a Gaussian kernel with standard deviation σ. The I_{GVC} image can be interpreted as vessel probability map (Fig. 4.13b). Gradient term calculated by I_{GVC} plays a significant role in curve stopping criteria. I_{GVC} can give a strong penalty to a curve when the curve tries to evolve through I_{GVC} boundaries.

Virtual gradient-assisted regional ACM

As presented above, a virtual gradient-assisted regional active contour model can be established by employing I_o, I_{GVC}, and $\nabla^* I_{VC}$:

$$\begin{aligned} F^{VGR}(c_1, c_2, \phi) = & \Big\{ \mu \int_{|\phi| \le \rho} \delta(\phi(x)) |\nabla \phi(x)| dx \\ & + \lambda_1 \int_{|\phi| \le \rho} |I_G(x) - c_1|^2 H(\phi(x)) dx \\ & + \lambda_2 \int_{|\phi| \le \rho} |I_G(x) - c_2|^2 (1 - H(\phi(x))) dx \\ & + \lambda_3 \int_{\phi} |I_{GVC}(x) - g_1|^2 H(\phi(x)) dx \\ & + \lambda_4 \int_{\phi} |I_{GVC}(x) - g_2|^2 (1 - H(\phi(x))) dx \Big\} \\ & \times \nu \cdot \mathcal{R}(|\nabla^* I_{VC}|), \end{aligned} \tag{4.21}$$

where $\mu \ge 0, \lambda_1, \lambda_2, \lambda_3, \lambda_4 > 0, \nu > 0$ are fixed parameters, $x \in \mathbb{R}^3$ is a spatial vector, I_G is Gaussian smoothed image from I_o, and

$$c_1(\phi) = \frac{\int_{|\phi| \le \rho} I_G(x) H(\phi(x)) dx}{\int_{|\phi| \le \rho} H(\phi(x)) dx}, \tag{4.22}$$

$$c_2(\phi) = \frac{\int_{|\phi| \le \rho} I_G(x) (1 - H(\phi(x))) dx}{\int_{|\phi| \le \rho} (1 - H(\phi(x))) dx} \tag{4.23}$$

are internal and external averages of I_G for narrow banded region defined by level set function ϕ and parameter ρ (Fig. 4.14). Φ forms approximate signed distance function while propagating so that ρ means the distance to current zero level set curve for each iteration. g_1 and g_2 are similarly defined with I_{GVC} image:

$$g_1(\phi) = \frac{\int_{\phi} I_{GVC}(x) H(\phi(x)) dx}{\int_{\phi} H(\phi(x)) dx}, \tag{4.24}$$

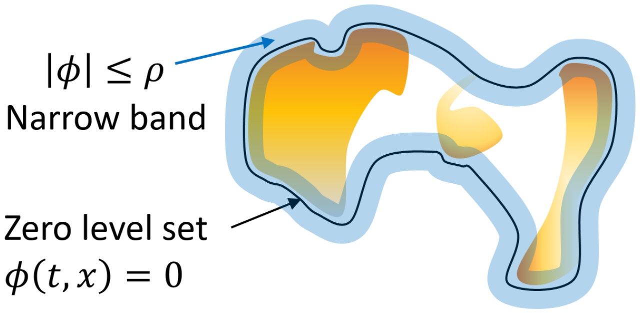


Figure 4.14: Narrow banded region of a level set function. The width of a band is defined by the ρ parameter. The image is visualized in 2D for simplicity.

$$g_2(\phi) = \frac{\int_\phi I_{GVC}(x)(1 - H(\phi(x)))dx}{\int_\phi (1 - H(\phi(x)))dx}. \tag{4.25}$$

The last term in (4.21) penalizes contour propagation at vessel boundaries by vessel probability map, I_{GVC}. R is a regularization function defined as

$$\mathcal{R}(x) = e^{-10x}. \tag{4.26}$$

A partial differential equation for this level set formulation can be defined similarly to [131]:

$$\begin{aligned} \frac{\partial \phi}{\partial t} =& \delta(\phi)\Big[\mu \cdot div(\frac{\nabla \phi}{|\nabla \phi|}) - \lambda_1 \cdot (I_G - c_1)^2 + \lambda_2 \cdot (I_G - c2)^2 \\ & - \lambda_3 \cdot (I_{GVC} - g_1)^2 + \lambda_4 \cdot (I_{GVC} - g_2)^2\Big] \times \nu \cdot \mathcal{R}(|\nabla^* I_{VC}|). \end{aligned} \tag{4.27}$$

The divergence of normalized gradient (i.e., $div(\frac{\nabla \phi}{|\nabla \phi|})$) is defined as a "*curvature*" which can be calculated by a level set function with respect to the mean curvature [165]:

$$k_M = div(\frac{\nabla \phi}{|\nabla \phi|}) = \frac{\begin{array}{r}(\phi_{yy} + \phi_{zz}){\phi_x}^2 + (\phi_{xx} + \phi_{zz}){\phi_y}^2 + (\phi_x x + \phi_y y){\phi_z}^2 \\ - 2\phi_x\phi_y\phi_{xy} - 2\phi_x\phi_z\phi_{xz} - 2\phi_y\phi_z\phi_{yz}\end{array}}{({\phi_x}^2 + {\phi_y}^2 + {\phi_z}^2)^{3/2}}. \tag{4.28}$$

And the Gaussian curvature also can be defined as [165]:

$$k_G = \frac{\begin{aligned}&{\phi_x}^2(\phi_{yy}\phi_{zz} - {\phi_{yz}}^2) + {\phi_y}^2(\phi_{xx}\phi_{zz} - {\phi_{xz}}^2) + {\phi_z}^2(\phi_{xx}\phi_{yy} - {\phi_{xy}}^2)\\ &\quad + \Big[\phi_x\phi_y(\phi_{xz}\phi_{yz} - \phi_{xy}\phi_{zz}) + \phi_y\phi_z(\phi_{xy}\phi_{xz} - \phi_{yz}\phi_{xx})\\ &\qquad + \phi_x\phi_z(\phi_{xy}\phi_{yz} - \phi_{xz}\phi_{yy})\Big]\end{aligned}}{({\phi_x}^2 + {\phi_y}^2 + {\phi_z}^2)^2}. \tag{4.29}$$

A narrow band regions were used for calculating c_1 and c_2 for three reasons: remove dependency with background distribution, get foreground distribution that is similar to minor vessel region, and get computational efficiency. The second reason is that minor vessel boundaries are easily ignored (i.e., curve evolves through the boundary) due to weak intensity contrast compared to the higher intensity distribution of internal major vessel areas. By calculating the narrow banded region of ϕ, minor vessels are more likely to be preserved by forming similar intensity distribution as the foreground.

To preserve fine details of vascular structure with many weak vessels, high smoothing with constant μ is not feasible (Fig. 4.15a). A smooth property while preserving fine details of a curve can be obtained based on the last term of (4.21) whereas relatively low smoothing constants are used (Fig. 4.15c). For initializing zero level set at time $t = 0$ (i.e., $\phi(0, x, y, z) = 0$), a dilated region of dense vessel candidate point cloud was used. Boundaries of the dilated region form the zero level set curve and ϕ is initialized by a signed distance map. The level set function was re-initialized for every 10 iterations. Due to the excellence of the initial contour, fast convergence of the level set function could be obtained. As mentioned above, narrow band optimization [129] is adopted to the proposed method (4.21).

The vesselness measures are not directly employed to the level set energy functional as presented in [135]. The main reason is that there is no global

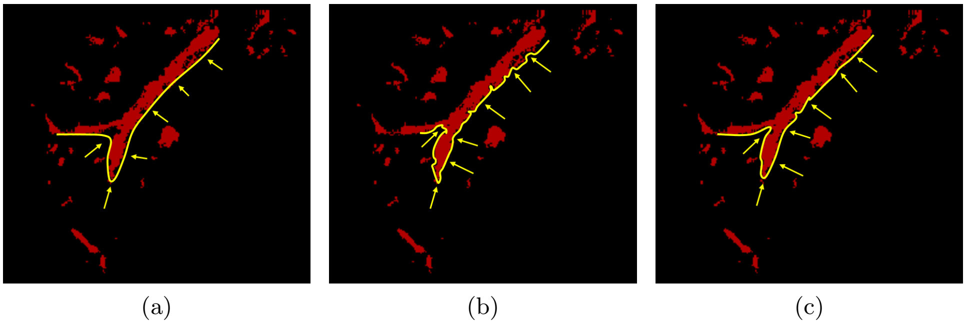

(a) (b) (c)

Figure 4.15: Conceptual active contour propagation results by (a) CV model with the higher smooth term (μ). (b) CV model with the higher regional term (λ_1, λ_2), and (c) the proposed model with virtual gradient assisted CV model (VGR) using both image and vessel candidates where red region represents dense vessel candidates.

optimum value of the scale parameter. That is, evolving contour at thick vessel region and thin vessel region must use different scale parameters in (2.34). Determining each scale parameter for each location is a very hard problem and using a maximum response with a unified large scale makes minor vessels hard to be detected. A vesselness measure metric is designed to aid contour propagation via a vessel probability map (Fig. 4.13b). Together with a robust analysis of regional intensity distribution, both tick and thin vessels can be successfully segmented.

4.3.2 Localized regional ACM

The previous section presented a level set formulation of the proposed method by modeling regions based on the global statistics (i.e., with the Chan-Vese model [131]). However, for heterogeneous objects, the global intensity distribution is not ideal even with narrow band optimizations. Inspired by Lankton and Tannenbaum [123], a localized analysis of regional terms is presented.

Localized regional energy term

Regional intensity statistics are a robust criterion for objects with homogeneous intensity distribution. However, it is infeasible for peripheral branches that are weakly represented in terms of low-intensity contrast and the presence of noise. A local region-based intensity statistics must be delivered for a fair discriminative competition of statistical intensity terms of energy functional. Thus, to compensate low-intensity contrast for weak vessel area, a local, regional analysis is employed for improving the global regional energy functional.

By introducing a characteristic function in terms of a radius parameter r, localized binary mask function can be defined [123]:

$$\mathcal{B}_r(x, y) = \begin{cases} 1, & ||x - y|| < r, \\ 0, & \text{otherwise}, \end{cases} \tag{4.30}$$

where $x, y \in \mathbb{R}^3$ are 3D spatial coordinates in Ω. The function will be 1 when the point y is within a ball of radius r centered at x, and 0 otherwise [123]. Figure 4.16 illustrates the activated $\mathcal{B}_r$ function in zero level sets (i.e., $\phi(t, x) = 0$). By employing a local region analysis, a contour can be propagated to the area of peripheral vessel branches that has low-intensity contrast. A new local-based regional term can be defined as

$$\int_{\phi_x} \delta(\phi(x)) \int_{\phi_y} \mathcal{B}_r(x, y) \cdot |I_G(y) - l_1(\phi(x))|^2 H(\phi(y)) dy dx \tag{4.31}$$

and

$$\int_{\phi_x} \delta(\phi(x)) \int_{\phi_y} \mathcal{B}_r(x, y) \cdot |I_G(y) - l_2(\phi(x))|^2 (1 - H(\phi(y))) dy dx, \tag{4.32}$$

where $x, y \in \mathcal{R}^3$. l_1 and l_2 are defined by internal and external regional statistics within local region constraint (i.e., $\mathcal{B}_r$). The formal expression will be presented in the following sub-section.

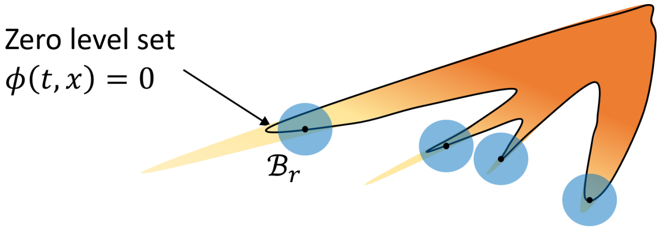


Figure 4.16: An illustration of localized regional energy term. Fading color illustrates the weak representation of intensity contrast in peripheral vessels. The localized region, i.e., $\mathcal{B}_r$, improves regional intensity statistics that can be unstable with global statistics. The zero level set in the figure is a schematic result of using simple narrow banded, global statistics.

Localized virtual gradient-assisted regional ACM

As presented in the previous sub-sections, a localized regional term can be integrated into the final active contour functional. The virtual gradient-assisted regional model can be extended based on localized region analysis (VGRL):

$$\begin{aligned} F^{VGRL}(l_1, l_2, \phi) = \Big\{ & \mu \int_{|\phi_x| \le \rho} \delta(\phi(x)) |\nabla \phi(x)| dx \\ & + \lambda_1 \int_{|\phi_x| \le \rho} \delta(\phi(x)) \int_{|\phi_y| \le \rho} \mathcal{B}_r(x, y) \cdot |I_G(y) - l_1(\phi(x))|^2 H(\phi(y)) dy dx \\ & + \lambda_2 \int_{|\phi_x| \le \rho} \delta(\phi(x)) \int_{|\phi_y| \le \rho} \mathcal{B}_r(x, y) \cdot |I_G(y) - l_2(\phi(x))|^2 (1 - H(\phi(y))) dy dx \\ & + \lambda_3 \int_{\phi} |I_{GVC}(x) - g_1(\phi(x))|^2 H(\phi(x)) dx \\ & + \lambda_4 \int_{\phi} |I_{GVC}(x) - g_2(\phi(x))|^2 (1 - H(\phi(x))) dx \Big\} \\ & \times \nu \cdot \mathcal{R}(|\nabla^* I_{VC}(x)|), \end{aligned} \tag{4.33}$$

where $x, y \in \mathbb{R}^3$. The difference from (4.21) are l_1 and l_2 that are defined by

$$l_1(\phi(x)) = \frac{\int_{|\phi_y| \le \rho} I_G(y) H(\phi(y)) \mathcal{B}_r(x, y) dy}{\int_{|\phi_y| \le \rho} H(\phi(y)) \mathcal{B}_r(x, y) dy}, \tag{4.34}$$

$$l_2(\phi(x)) = \frac{\int_{|\phi_y|\leq\rho} I_G(y)(1 - H(\phi(y)))\mathcal{B}_r(x,y)dy}{\int_{|\phi_y|\leq\rho}(1 - H(\phi(y)))\mathcal{B}_r(x,y)dy}, \tag{4.35}$$

where x and y are spatial vectors same as in (4.33) and function $\mathcal{B}_r$ is defined by radius parameter r (4.30).

Intensity averages in the interior and exterior of the curve are localized by proximity function $\mathcal{B}_r$ and width of narrow band ρ. Partial differential equation of (4.33) can be easily formulated by setting l_1 and l_2 as constants:

$$\begin{aligned}\frac{\partial\phi}{\partial t}(x) =&\delta(\phi(x))\Bigg[\mu\cdot div(\frac{\nabla\phi(x)}{|\nabla\phi(x)|})\\ &-\lambda_1\int_{|\phi_y|\leq\rho}\mathcal{B}_r(x,y)\delta(\phi(y))\Big((I_G(y) - l_1(\phi(x)))^2\Big)dy\\ &+\lambda_2\int_{|\phi_y|\leq\rho}\mathcal{B}_r(x,y)\delta(\phi(y))\Big((I_G(y) - l_2(\phi(x)))^2\Big)dy\\ &-\lambda_3\Big(I_{GVC}(x) - g_1(\phi(x))\Big)^2\\ &+\lambda_4\Big(I_{GVC}(x) - g_2(\phi(x))\Big)^2\Bigg]\times\nu\cdot\mathcal{R}(|\nabla^* I_{VC}(x)|).\end{aligned} \tag{4.36}$$

In VGRL model (4.33), localized distance was used for region term calculation. Calculating regional intensity distribution with proximity constraints makes a better approximation of foreground weak vessel intensity distribution.

4.4 Experimental results

4.4.1 Overview

In the experiments, 2D segmentation methods are firstly presented in detail. Subsequently, a comparison among the proposed method and the other active contour models are demonstrated under the same condition of an initial contour. Geodesic active contour (GAC) [130], Chane-Vese model (CV) [131], vascular active contour (VAC) [135], and the proposed models are used for comparison. Volumetric validation with manual segmentation results is not presented because annotated data is highly expert-dependent (i.e., inter-observer variability) and hard to acquire golden standard segmentation results. Instead, proper validation is performed for the proposed method by using vessel tree branching points together with 3D visualization of the segmented object and 2D visualization of example axial slices. Clinical experts manually identified numerous bifurcation points for quantitative analysis of the proposed method.

4.4.2 Data configurations and environment

The dataset includes 55 abdominal CT images and the corresponding annotated points (i.e., bifurcation points) by clinical experts. For the training data, 5 images were used to optimize the parameters of the proposed method. The other 50 images were used for quantitative evaluations of liver vessel segmentation. In the dataset, a slice thickness ranged from 0.6mm to 2.0mm and pixel sizes ranged from 0.7mm to 1.0mm. Portal phase CT images were used due to the high contrast of the vessel region. All experiments were tested on an Intel i7-6700K desktop system with a 4.0GHz processor, 16GB of memory, and Nvidia Titan X (Pascal) GPU machine.

4.4.3 2D segmentation

To get robust segmentation result in 2D maximum intensity slab images, the analysis of a multi-scale vessel enhancement filter [102] responses with and without BM3D [151] denoising was performed. Figures 4.17 and 4.18 shows the difference of vesselness filter responses between the original image (i.e., maximum intensity slab image) and BM3D-denoised image. All experiments have same vesselness function with fixed parameter $\beta = 0.5$ and $c = 0.5$ in (4.10). Experiments show that low sigma value with $1 \leq \sigma \leq 3$ is the best scale for fine vessel enhancement (Figs. 4.17b and 4.18b). Including higher sigma

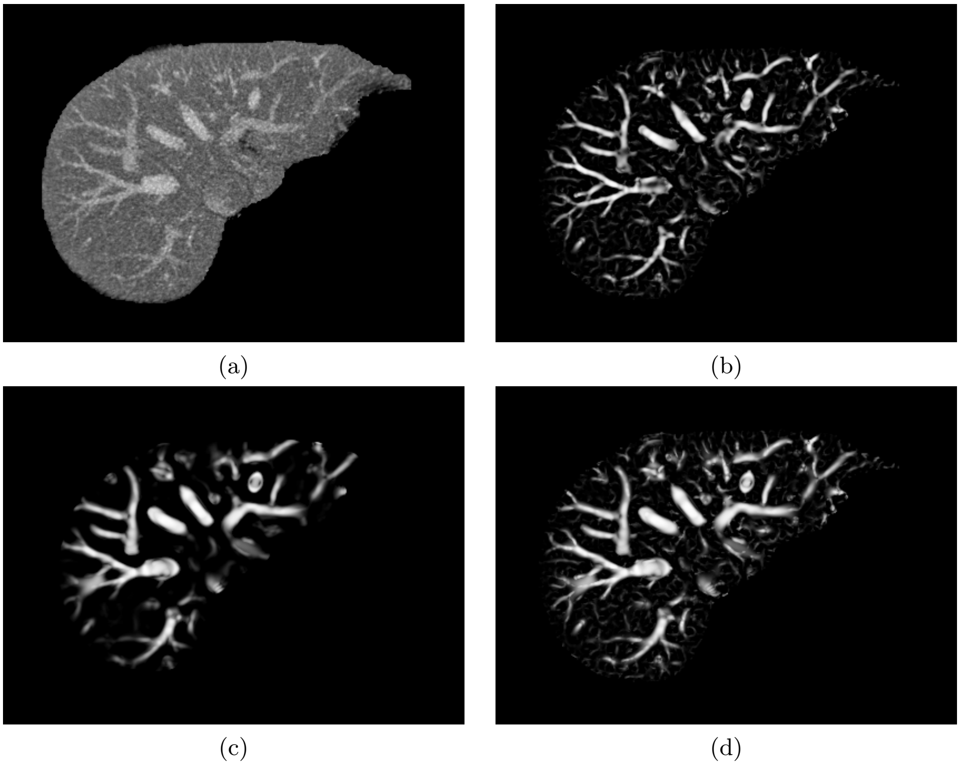
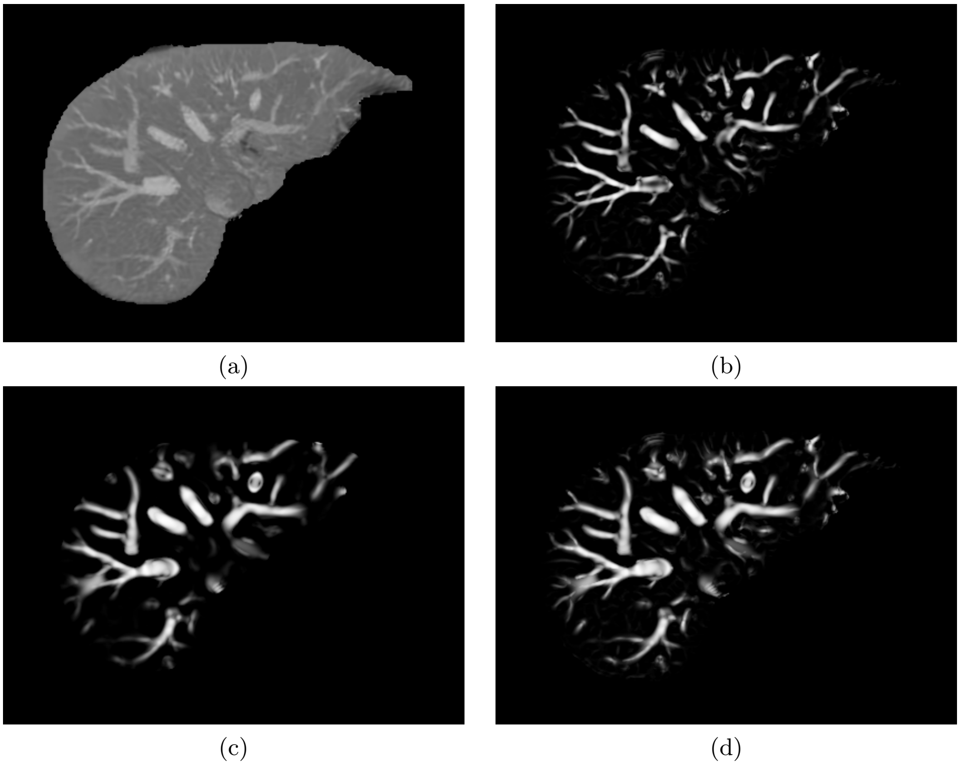


Figure 4.17: Results of multi-scale vessel enhancement filtering [102] responses on various scale parameters applied to MIS image without BM3D denoising [151]. (a) The original maximum intensity slab (MIS) image with z-axis direction with 7 slab thickness. (b)-(d) shows vesselness filter results from (a). Each corresponds to multi-scale parameters, σ (standard deviation of the Gaussian kernel). (b): $1 \leq \sigma \leq 3$, (c): $4 \leq \sigma \leq 6$, and (d): $1 \leq \sigma \leq 6$, respectively.

regard clustered minor vessel region as a one thick vessel (Figs. 4.17d and 4.18d). The range of low sigma values was adopted for vessel enhancement because the main purpose of generating maximum intensity slab images and performing vesselness filtering is to get precise vessel candidates at minor vessel regions. For the use of low scale sigma value, the BM3D [151] must be applied before vesselness filtering because Gaussian kernel with low sigma values does not suppress noise effectively compared to the higher ones (Figs. 4.17 and 4.18).


(a)

(b)

(c)

(d)

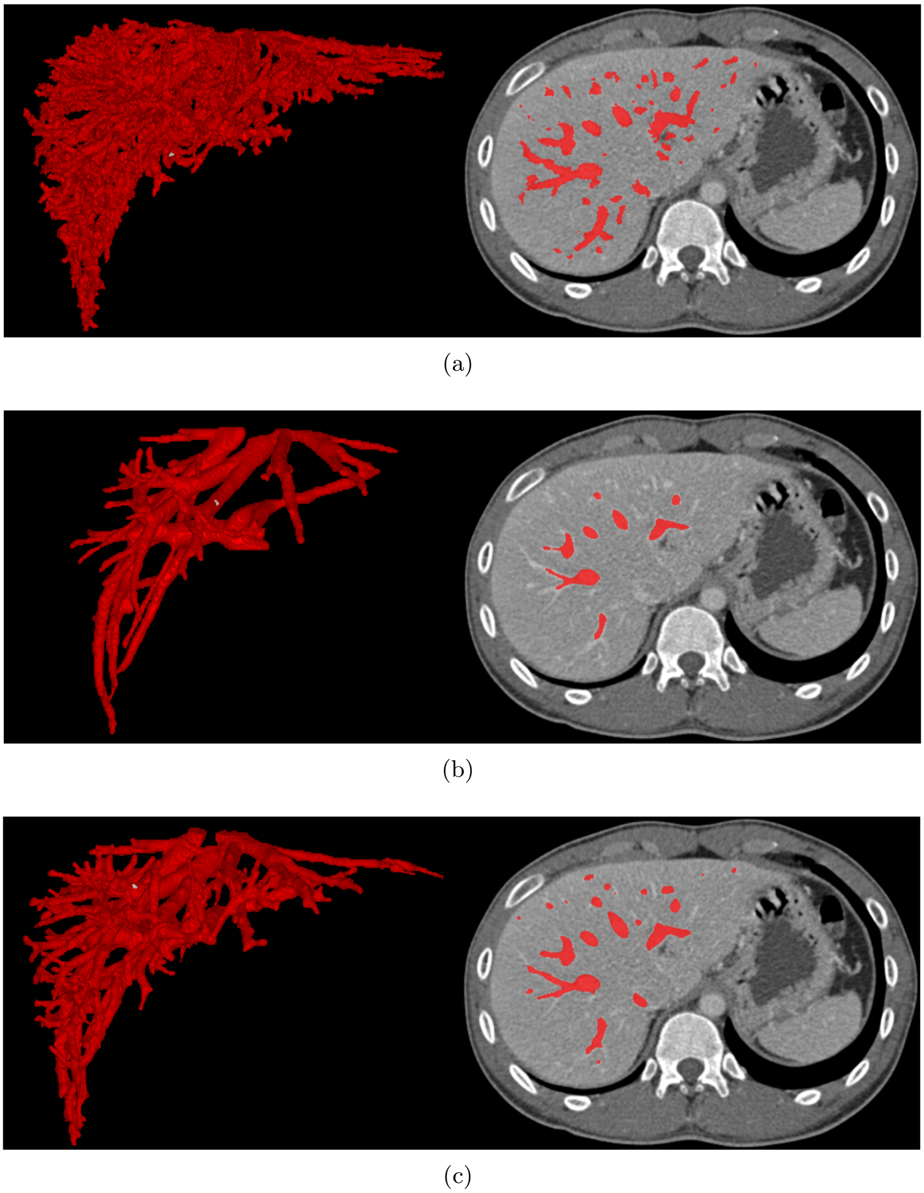
Figure 4.18: Results of multi-scale vessel enhancement filtering [102] responses on various scale parameters applied to MIS image with BM3D denoising [151]. (a) The BM3D denoised maximum intensity slab (MIS) image with z-axis direction with 7 slab thickness. (b)-(d) shows vesselness filter results from (a). Each corresponds to multi-scale parameters, σ (standard deviation of the Gaussian kernel). (b): $1 \leq \sigma \leq 3$, (c): $4 \leq \sigma \leq 6$, and (d): $1 \leq \sigma \leq 6$, respectively.

Fixed 7-voxel thickness is used for all maximum intensity slab image generation and a single-voxel sized shift is performed for each direction (Fig. 4.11). For the final segmentation, 0.4 value of threshold was fixed. The threshold value was set by relatively high and fixed value because the lower value can lead to over-segmentation that might be noise in the vessel candidate set. In the fact that maximum intensity slab image generation is performed by single-voxel sized shift with three axis-aligned directions, the unsegmented region is a single image can be compromised by other maximum intensity slab images. It was experimentally observed that the unsegmented peripheral branch vessel region, represented by line structure in a certain maximum intensity slab image, is successfully segmented by blob structure in other directional images.

In the experiments, maximum intensity slab images were generated by three axis-aligned directions because of the computational efficiency. One might consider adding additional oblique directional projections to get more robust and dense vessel candidate generation. However, the computational cost of the maximum intensity slab image segmentation is relatively high compared with the other procedures and three axis-aligned directions were suffice based on observations.

4.4.4 ACM comparisons

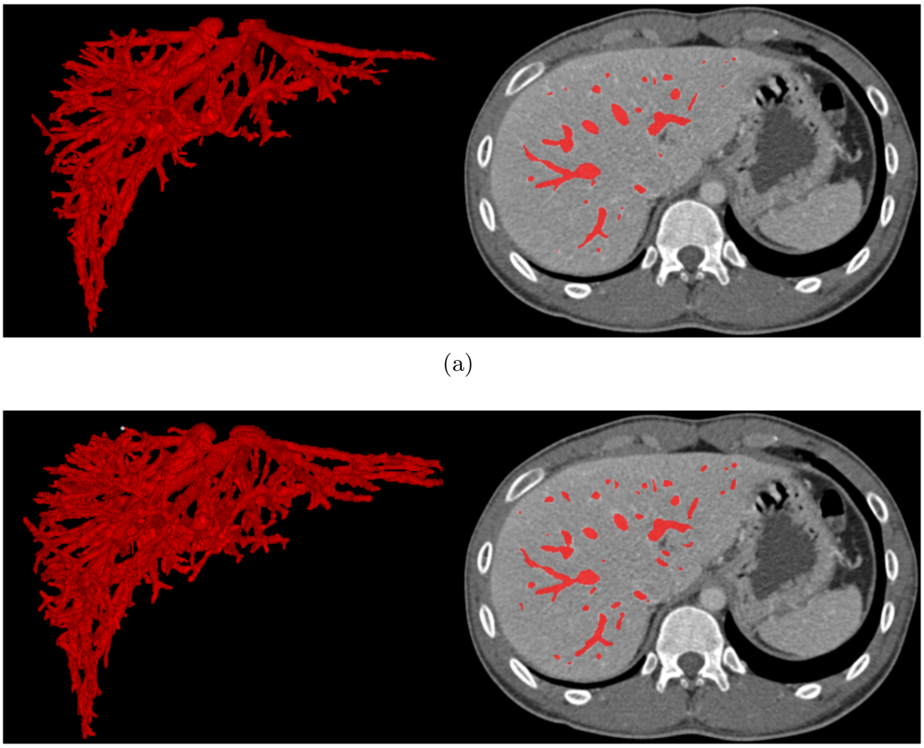
Figures 4.19 and 4.20 shows vessel segmentation results using different active contour models. The same initial contour was used based on the proposed method via dense vessel candidates. With the help of automatic good initial condition, all experimented active contour models showed good results. However, other models have several limitations compared to the proposed methods. For GAC, result is noisy and boundaries are irregular Figs. 4.19a and 4.21b. This is because GAC only uses the edge information from the original noisy image that makes it hard to detect minor vessel edges only by gradient. The parameter $\nu = 1.0$ was used for the equation (2.43). In the segmentation results of CV method, too many fine vessels are lost even with the good initial condition Figs. 4.19b and 4.21c. This result shows that the vessel region consisting of both high and low contrast vessels is very hard to estimate the global intensity distribution of the internal area. The parameter values $\mu = 0.5, \lambda_1 = 0.001$, and $\lambda_2 = 0.001$ were used in the equation (2.50). VAC model segmented more accu-


(a)

(b)

(c)

Figure 4.19: Liver vessel segmentation results with several active contour models (ACMs). The first column shows 3D object visualization of extracted vessel regions and the second column shows the example axial slices of 2D segmentation results. Each row represents different ACMs: (a) GAC [130], (b) CV [131], and (c) VAC [135], respectively.

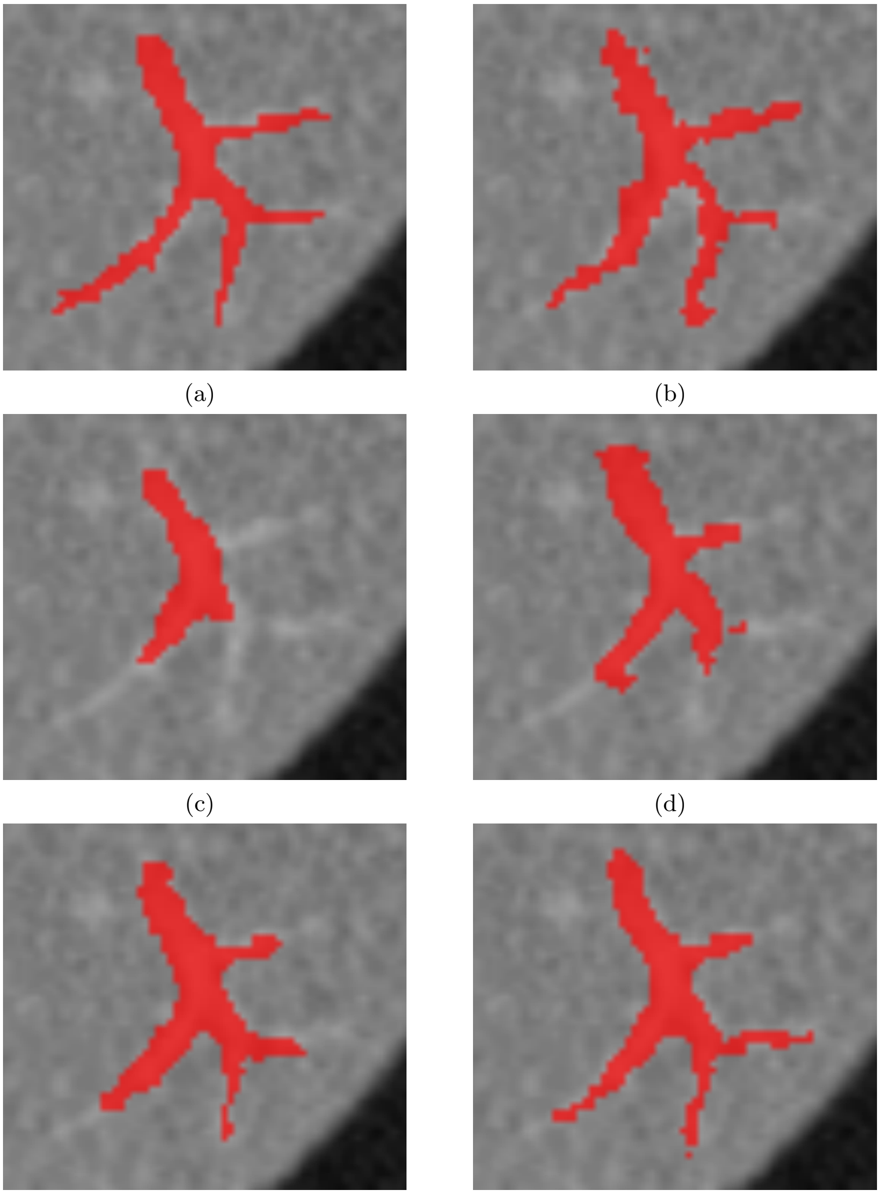


(a)

(b)

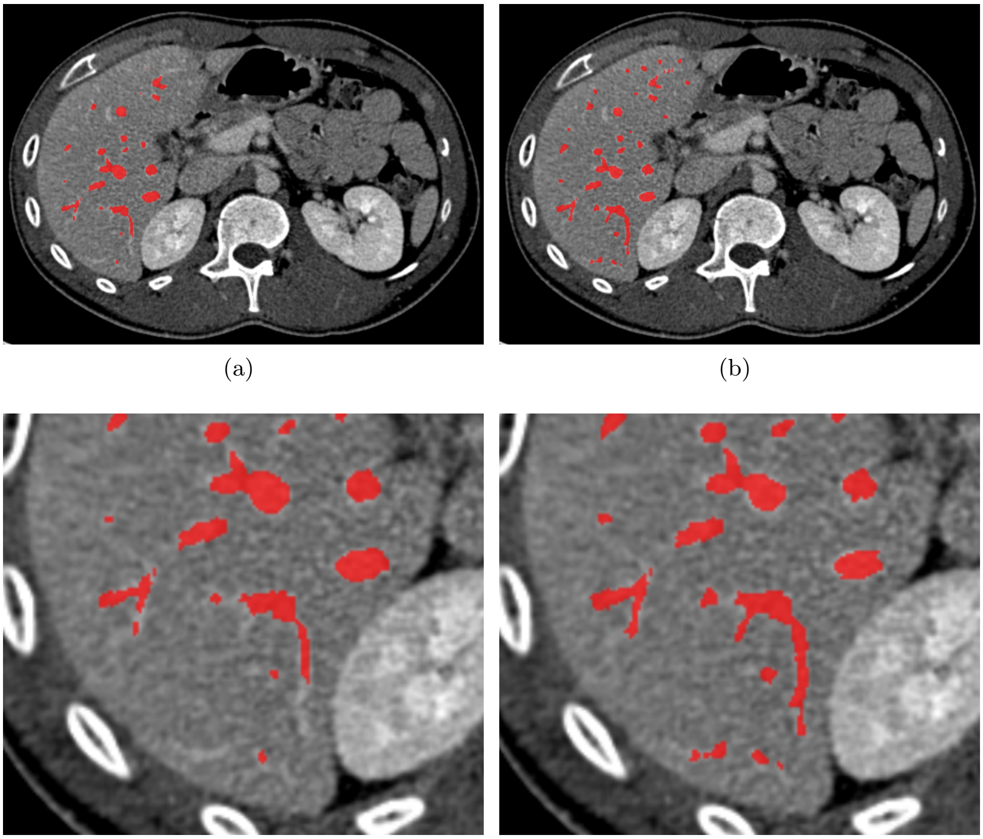
Figure 4.20: Liver vessel segmentation results with the proposed active contour models. The first column shows 3D object visualization of extracted vessel regions and the second column shows the example axial slices of 2D segmentation results. (a) VGR and (b) VGRL.

rate vessel region than CV model (Figs. 4.19c and 4.21d) due to vascular vector field energy. The parameter values were set as the same values in the original paper [135]. Figures 4.20a and 4.21e show the proposed VGR model results that use image distribution from both original image and the generated vessel probability map together with virtual gradients. As clearly demonstrated, VGR model successfully segment vessel region with smooth boundaries and fine details. The parameter values $\mu = 0.5, \lambda_1 = 0.001, \lambda_2 = 0.001, \lambda_3 = 10$, and $\lambda_4 = 10$ were used in the equation (4.27). λ_3 and λ_4 are relatively large because



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

Figure 4.21: Comparison of thin vessel segmentation results with several active contour models. (a) Manual ground-truth in local region, (b) GAC [130], (c) CV [131], (d) VAC [135], (e) VGR, and (f) VGRL, respectively.



(a)

(b)

(c)

(d)

Figure 4.22: Liver vessel segmentation results by the proposed VGR and VGRL models. The first and second columns represent VGR and VGRL, respectively.

of the range of an image. For the detection of more fine details of weak vessels, VGRL model is experimented as presented in (4.33).

The localized region calculation leads to accurate estimation of regional intensity distributions for each position. Thus, minor vessels are far more preserved than VGR model by estimating intensity distribution in local rather than global region (Figs. 4.20b and 4.21f). Moreover, the boundaries of vessel are more accurately delineated due to the same effect of estimating a local intensity

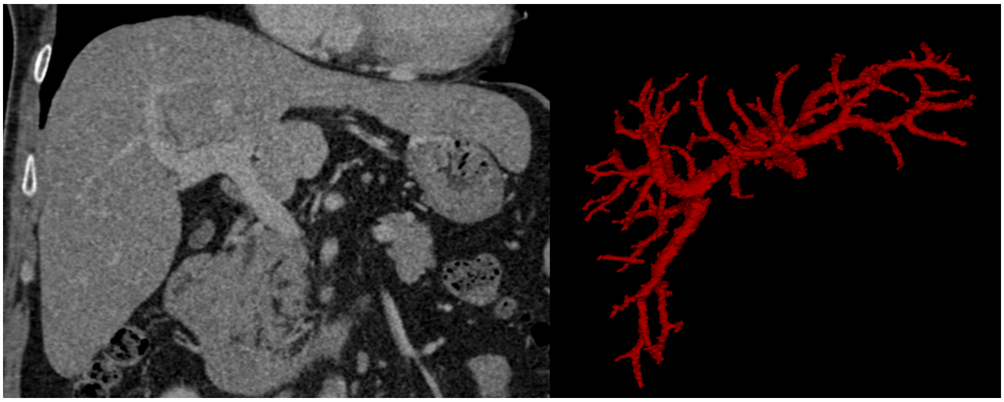


Figure 4.23: An example result of the proposed VGRL active contour model.

distribution. The parameter $\mu = 0.5, \lambda_1 = 0.001, \lambda_2 = 0.001, \lambda_3 = 10, \lambda_4 = 10,$ and $r = 10$ were used in the equation (4.36). All active contour model experiments made convergence within 100 iterations.

4.4.5 Evaluation of bifurcation points

For quantitative evaluation of the proposed method, the number of bifurcation points (i.e., branching points of vessel tree structures) was compared. Clinical experts were asked to identify true vessel branching points manually in each of the fifty datasets. In this way, the manual points which serve as the ground-truth for the accuracy assessment could be obtained. Bifurcation points of the active contour models' results were automatically generated by localizing branching regions of segmented vessel tree skeletons (Fig. 4.24). The skeleton is first extracted from segmented vessel tree via distance ordered thinning-based methods [166–168]. The center-lined skeleton voxels were each classified using neighborhood connectivity criteria: 1-connected voxels as *end*, 2-connected voxels as *line*, and 3 or more connected voxels as *branch*. Finally, the branching points are localized as a center of connected *branch* voxel clusters. The accuracy

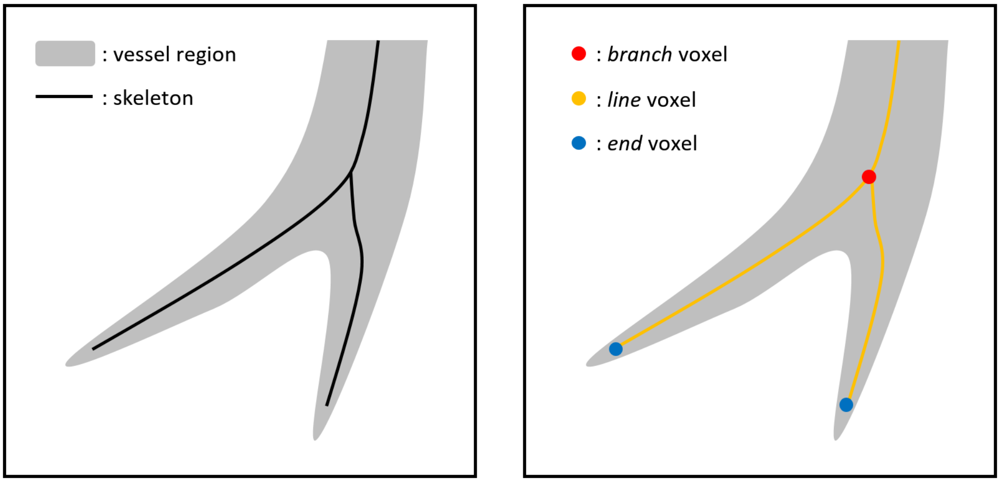


Figure 4.24: Liver vessel tree skeletonization and classification of skeleton voxels. The black centered line of the left image represents vessel tree skeleton. The right image describes *branch*, *line*, and *end* voxel classification using connectivity criterion. *Line* voxel region is represented by a curve for simplicity.

is evaluated in terms of two different factors, as follows:

$$E_{fp} = \frac{num\{B_a\} - num\{B_a \cap B_m\}}{num\{B_m\}}, \tag{4.37}$$

$$E_{fn} = \frac{num\{B_m\} - num\{B_a \cap B_m\}}{num\{B_m\}}, \tag{4.38}$$

where B_a is automatically detected branching point and B_m is a manually identified branching point by a clinical expert. For constructing the set, $\{B_a \cap B_m\}$, the one closest corresponding B_a for each B_m was mapped if and only if the Euclidean distance between them is less than 1mm to their voxel positions. Then, if the condition is met, the voxel was added to the set. The false positive error, E_{fp}, is the ratio of the set of B_a but not in the set of B_m to the set of B_m. The false negative error, E_{fn}, is the ratio of the set of B_m but not in the set of B_a to the set of B_m.

Tables 4.1 and 4.2 represent the number of branching nodes (i.e., bifurcation points) of vessel tree, false positive, and false negative scores on five training

Table 4.1
Number of branching nodes of the vessel tree (five training images).

Dataset	**GAC** [130]	**CV** [131]	**VAC** [135]	**VGR**	**VGRL**	***Manual***
1	923	81	216	274	515	536
2	901	80	215	250	487	529
3	876	76	201	258	502	541
4	943	80	207	269	509	533
5	884	77	205	261	496	521

Table 4.2
Accuracy assessment results of vessel segmentation (five training images).

Dataset	**Error**	***GAC*** [130]	***CV*** [131]	***VAC*** [135]	***VGR***	***VGRL***
1	E_{fp}	0.761	0.002	0.002	0.002	0.015
	E_{fn}	0.039	0.851	0.599	0.490	0.054
2	E_{fp}	0.775	0.000	0.008	0.013	0.004
	E_{fn}	0.072	0.849	0.601	0.541	0.083
3	E_{fp}	0.661	0.004	0.008	0.013	0.004
	E_{fn}	0.043	0.863	0.636	0.536	0.076
4	E_{fp}	0.814	0.000	0.015	0.004	0.019
	E_{fn}	0.045	0.850	0.627	0.499	0.064
5	E_{fp}	0.735	0.002	0.004	0.004	0.002
	E_{fn}	0.038	0.854	0.610	0.503	0.050
average	E_{fp}	**0.749**	**0.001**	**0.007**	**0.007**	**0.009**
	E_{fn}	**0.047**	**0.853**	**0.615**	**0.514**	**0.065**

images. Table 4.1 shows the number of bifurcation points on ground-truth annotations. Table 4.2 summarizes the errors of the five dataset. The average number of elements in a set $\{B_a \cap B_m\}$ were 507, 78, 205, 259, and 487 for GAC, CV, VAC, VGR, and VGRL, respectively. The proposed VGR model segmented 181 and 54 more true branching nodes than that of CV and VAC models on average. VGRL model segmented even more correct nodes than the

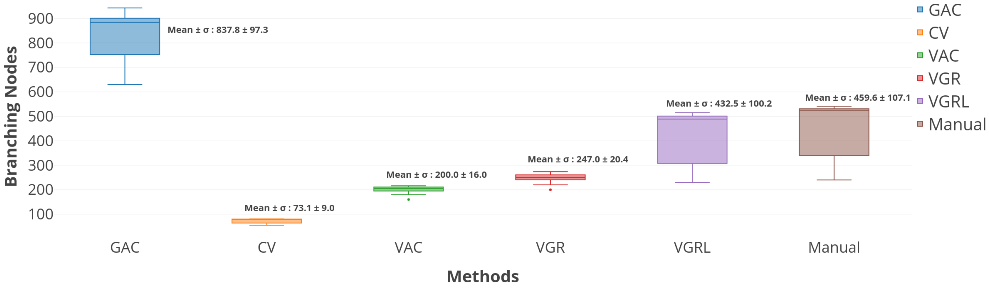


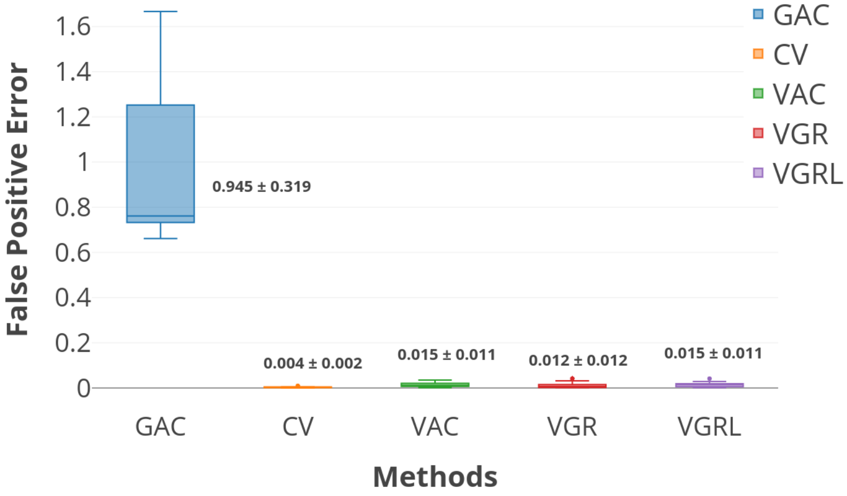
Figure 4.25: Number of branching nodes of the vessel trees with respect to each active contour model method and manually annotated points.

VGR model that is 228 on average.

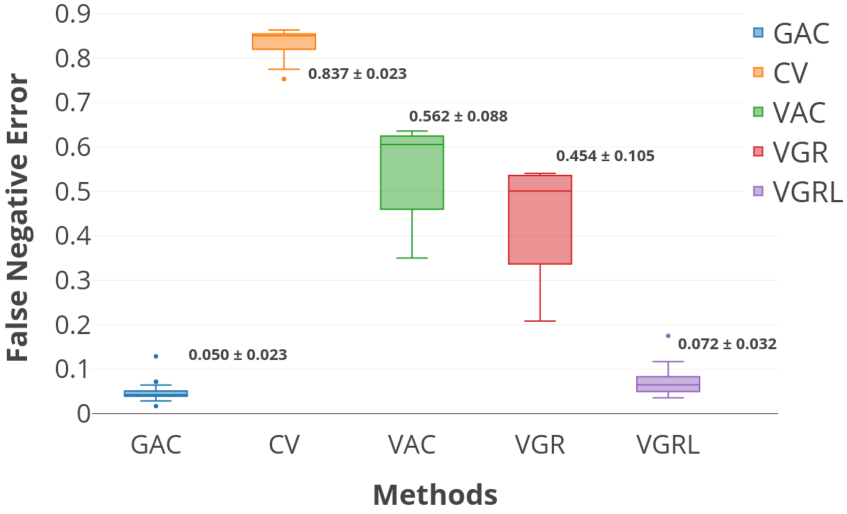
Figure 4.25 shows the number of detected branching points by active contour models and the manually annotated points. The observation is that the VGR and VGRL methods were more successful in minor vessel segmentation tasks than CV and VAC models. Numerous branching nodes of the GAC method is mainly due to the noise of the segmented vessel region.

Figure 4.26 summarizes the errors on the fifty datasets. The average number of elements in a set $\{B_a \cap B_m\}$ were 437, 72, 194, 242, and 427 for GAC, CV, VAC, VGR, and VGRL, respectively. The VGR model segmented 170 and 48 more true branching nodes than that of CV and VAC models on average. VGRL model segmented even more correct nodes than the VGR model that is 185 on average.

In all datasets, E_{fp} was relatively high in GAC method, indicating that GAC detected many false bifurcation points (Fig. 4.26a). Even with low value of E_{fn} (Fig. 4.26b), GAC method was not successful due to many false positive detections. In the case of CV, the average value of E_{fp} was the lowest, which was 0.004 but E_{fn} marked the highest, indicating that the CV method failed to segment thin vessel region. The proposed VGR and VGRL models apparently



(a)



(b)

Figure 4.26: Assessment of (a) False positive error (E_{fp}) and (b) false negative error (E_{fn}) for each active contour model method.

showed the best results among other methods. VGR model resulted in lower E_{fn} and E_{fp} compared to the VAC model that is better detection of minor vessels without extra false positive error. VGRL model detected even more minor vessels than VGR that is close to the manual detection along with a slight increase of E_{fp}.

The choice of methods between (4.21) and (4.33) depend on time complexity and accuracy. If time performance is not critical and more accurate fine details of minor vessels are required, (4.33) will be a better method. Equation (4.21) is relatively more efficient than (4.33) and still preserves minor vessels well compared to other models.

4.4.6 Computational performance

For the evaluation of the computational performance of the proposed method, the two major steps were measured: generation of dense vessel candidates and clustering of dense vessel candidates using an active contour model. The processing time for dense vessel candidate generation, averaged over multiple tests for all the dataset, was 90s tested under full GPU implementations. It took 31s, 24s, 119s, 40s, and 167s on average for the active contour model segmentation via GAC, CV, VAC, VGR, and VGRL, respectively. For the whole processing (i.e., combined VC generation and active contour propagation), it took 121s, 114s, 209s, 130s, and 257s on average for each patient's CT volume via GAC, CV, VAC, VGR, and VGRL, respectively. All active contour model segmentation procedures were implemented on a CPU basis. The typical data size for a computational performance evaluation was $512 \times 512 \times 240$ with an in-plane pixel spacing of 0.6mm and a slice thickness of 0.7mm. The proposed VGR method was faster and accurate than the VAC method. VGRL model was the most successful method in minor vessel segmentation problem. However, the

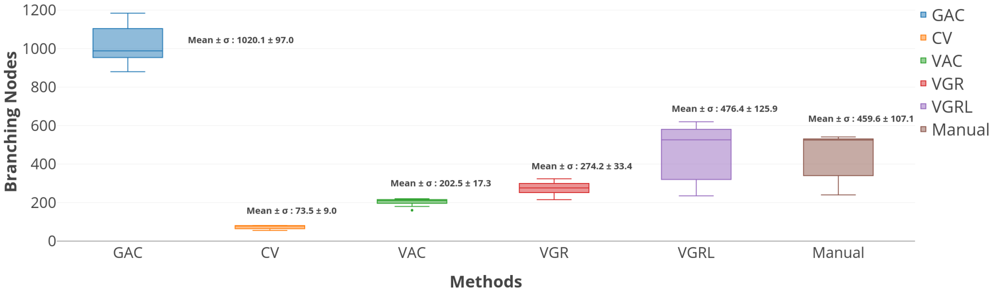
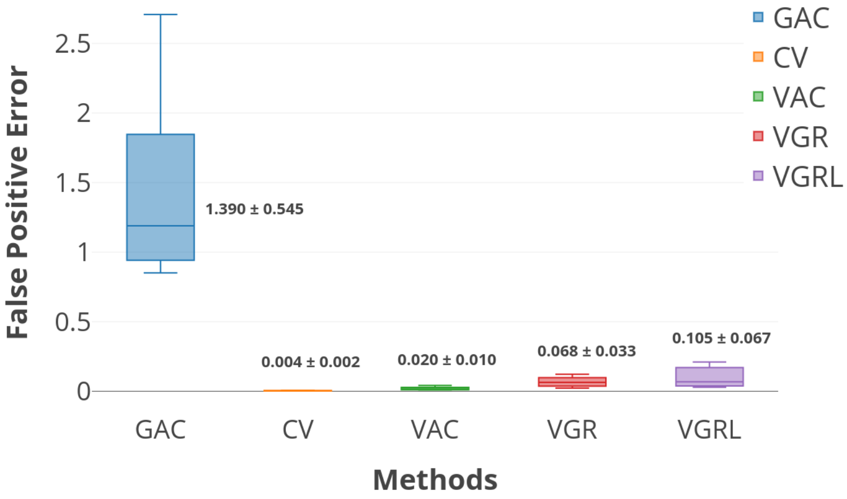


Figure 4.27: Number of branching nodes of the vessel trees with respect to each active contour model method and manually annotated points (without slab).

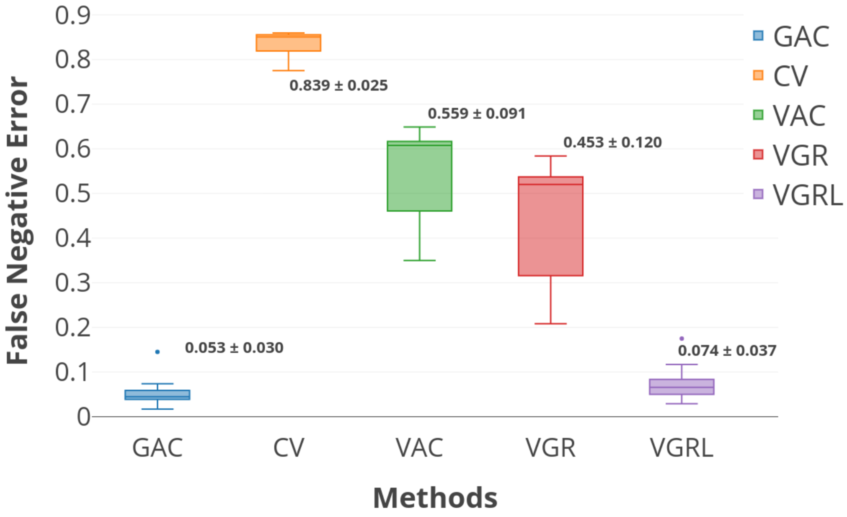
computational efficiency of the method was the worst because of the local region analysis for all contour positions to propagate.

4.4.7 Ablation study

The effect of maximum intensity slab images has additionally experimented. As illustrated in Fig. 4.9, 2D segmentation without maximum intensity slab images (i.e., using a 1-voxel thickness plane) leads to an inaccurate and insufficient vessel candidate generation. Without the help of robust and dense vessel candidates, the proposed method is not able to segment a vessel region accurately because the proposed active contour models (i.e., VGR and VGRL models) are mainly dependent to I_{GVC} image which is generated from vessel candidates ((4.21) and (4.33)). Figures 4.27 and 4.28 show results of active contour models using vessel candidates that is generated without slab. The number of branching nodes of a vessel tree was globally increased in Fig. 4.27 compared to that of with slab images (Fig. 4.25). The results of the CV and VAC model had relatively no significant difference because the initial contour did not affect their final results compared to other methods. However, the number of branching nodes of vessel tree with GAC, VGR, and VGRL models (Fig. 4.27) was



(a)



(b)

Figure 4.28: Assessment of (a) False positive error (E_{fp}) and (b) false negative error (E_{fn}) for each active contour model method (without slab).

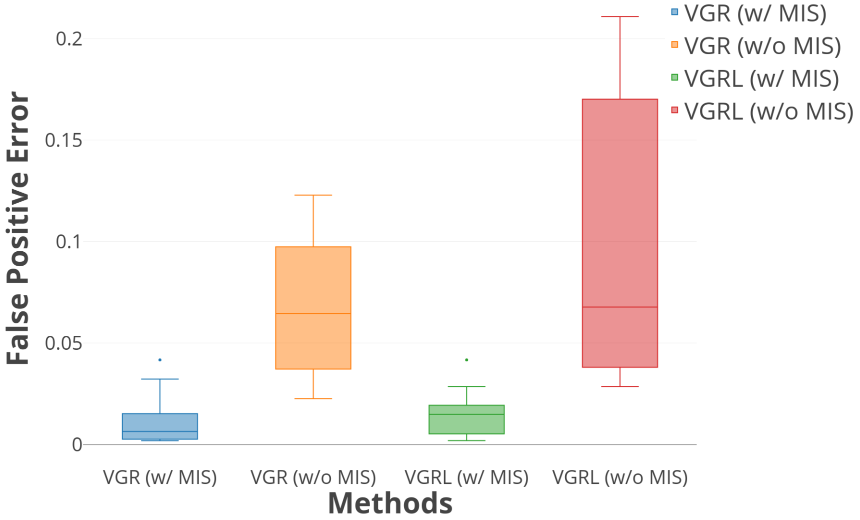
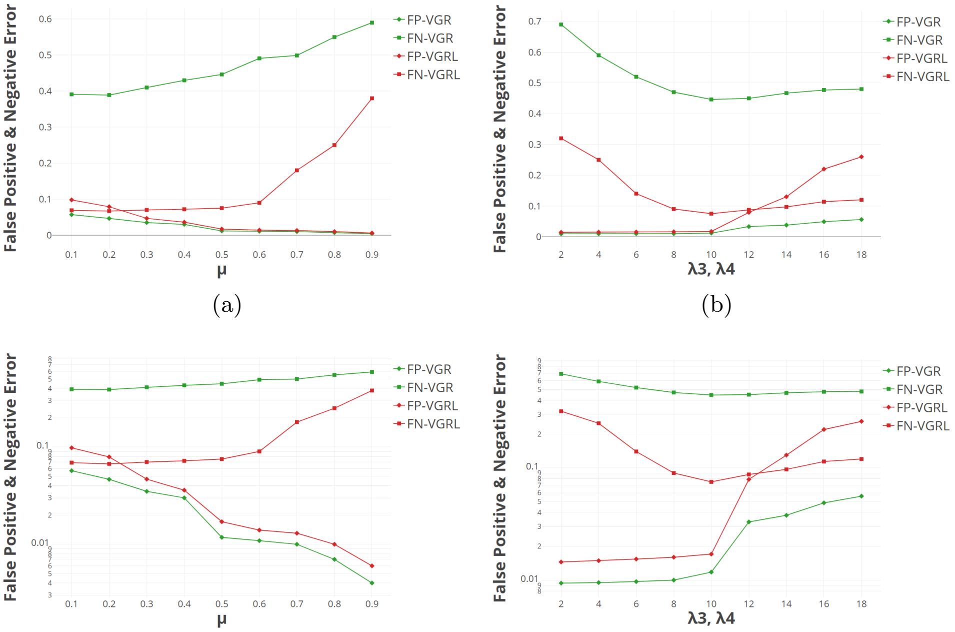


Figure 4.29: False positive error (E_{fp}) assessment for VGR and VGRL models with and without maximum intensity slab images.

increased with the simultaneous increase of E_{fp} (Fig. 4.28a). Figures 4.27 and 4.28a clearly indicate that a lot of false detection are presented with inaccurate vessel candidates which are generated by 2D image segmentation without thickness. False negative error (E_{fn}) in Fig. 4.28b shows no significant difference. Even if E_{fn} is decreased slightly, it is not considered as a performance gain because of the simultaneous increase of E_{fp}. For better and clear visualizations, Fig. 4.29 shows the difference with or without slab image with respect to false positive errors.

4.4.8 Parameter study

The performance of the active contour model algorithm is sensitive to parameters. For the optimal set of parameters, 5 CT images were used to train the parameters. The smoothness of a zero level set is controlled by the μ parameter. In Fig. 4.30, it is evident that a higher value of μ loses minor vessel branches



(a) (b)

(c) (d)

Figure 4.30: Study of parameters (a) μ and (b) λ_3, λ_4. (c) and (d) is log-scaled plotting for (a) and (b), respectively. Errors are averaged for 5 CT images.

(i.e., high false negative error) and lower value of μ makes noisy boundaries (i.e., high false positive error). In (4.21) and (4.33), the degree of influence of image intensity distribution and vessel candidate distribution is controlled by λ_1, λ_2, and λ_3, λ_4, respectively. That is, setting relatively higher values of λ_1, λ_2 makes the level set function evolve more dependent on image intensity distribution rather than vessel candidate distribution. On the contrary, vessel candidate distribution dominantly affects the evolution of level set function with relatively higher values of λ_3, λ_4. The two distribution must mutually coexist and balanced. As clearly indicated in Fig. 4.30, lower values of λ_3 and λ_4 increase false negative errors. the main reason for false negative errors is that minor vessel region is hard to be determined in the original image intensities. In

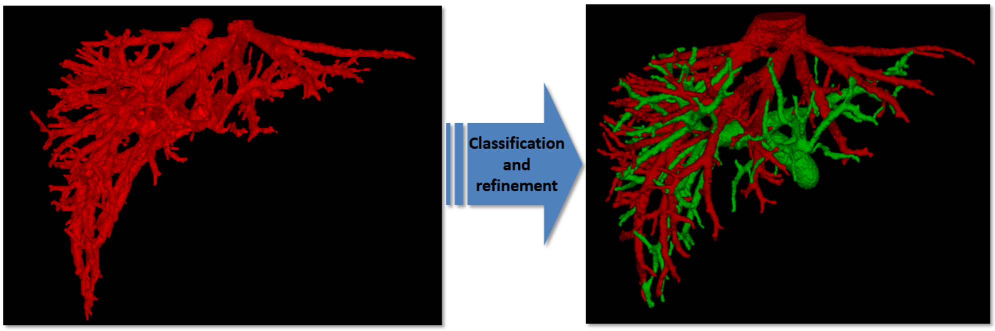


Figure 4.31: Classification of hepatic and portal veins (i.e., separation).

case of higher values of λ_3 and λ_4, the false positive errors increased indicating that it failed to segment smooth vessel structure.

There are two undiscussed parameters regarding slab image generation: slab thickness and shift interval. Increasing slab thickness and shift interval values significantly decrease the density of the vessel candidate set. The reason for the effect of the slab thickness parameter is that when generating maximum intensity slab images, different vessel regions overlap and vessels that have relatively lower intensities than others are more easy to be ignored. The use of a large thickness makes more overlapping vessel regions in the image and that means more information loss (i.e., loss of vessel candidates). On the contrary, too small thickness value lowers the benefit of a maximum intensity image that is a reduction of noise variance. The 7-voxel thickness was determined by numerous experiments because it maximizes both the image quality and density of the vessel candidate set. For other applications, the two parameters might be set higher for the automatic extraction of coarse seed points.

4.5 Application to portal vein analysis

The proposed vessel segmentation technique has two major advantages: extracts all existing vessel segments that are contrast-enhanced regardless of connectiv-

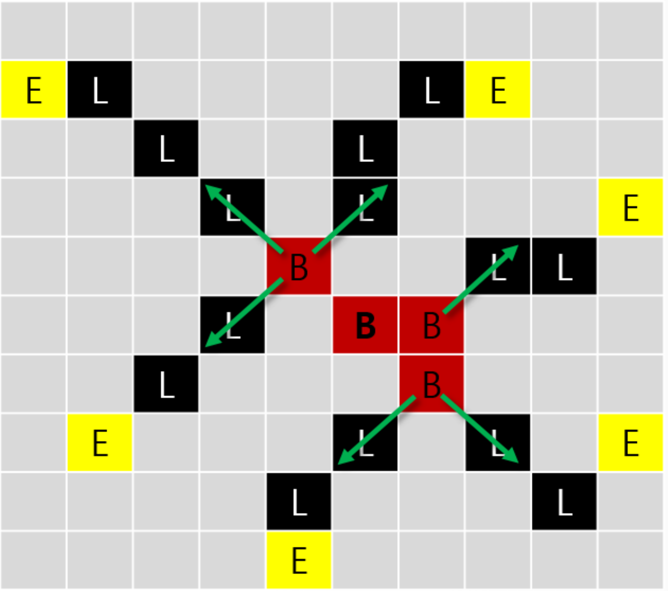


Figure 4.32: Skeleton voxels. The skeleton voxels are classified using neighborhood connectivity criteria: 1-connected voxels as *end*, 2-connected voxels as *line*, and 3 or more connected voxels as *branch*. The branching points are localized as a center of connected *branch* voxel clusters (bold 'B' in the figure). The propagation is performed on a *branch* voxel basis.

ity and segments weak peripheral branches. In this section, a method of separating hepatic and portal veins is introduced for further portal vein structural analysis (Fig. 4.31). The structure of the portal vein is an important clinical measure that anatomically divides (i.e., partitions) liver sections that provides accurate surgical planning such as liver resections.

Once well contrast-enhanced vessels are segmented, the region is typically connected (i.e., fully connected as a one object). To separate portal vein regions from the hepatic vein, a skeleton-based confident flow method is presented. First, the object is processed under morphological erosion operation which shrinks an object. By erosion, the object is separated into multiple segments. The two largest segments can be easily obtained by connected com-

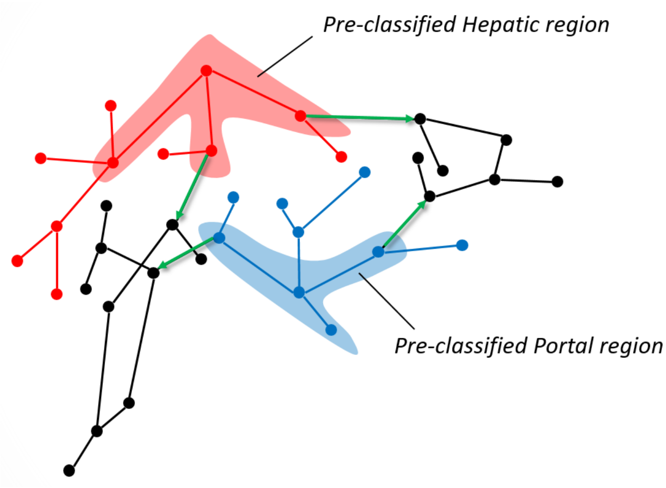
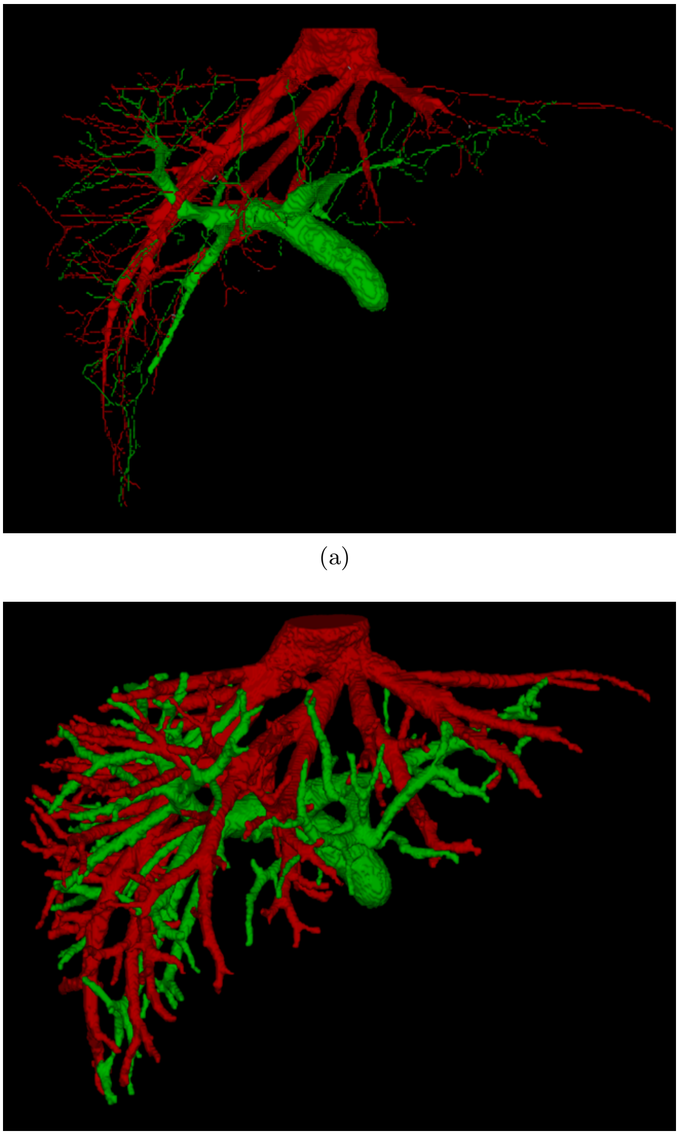


Figure 4.33: Classification of hepatic and portal veins. The red region, lines, and points indicate the pre-classified hepatic region. The local branches can be automatically classified without conflict. Similarly, the blues indicate the portal region. The two skeletons are propagated to classify the remaining branches.

ponent analysis [149], therefore, rough rooted regions of hepatic and portal veins can be obtained. From the pre-classified hepatic and portal vein regions, a skeleton can be propagated based on tree structures (Figs. 4.32 and 4.33).

The skeletonization is performed based on distance-ordered, topology preserving thinning-based method [169–172]. After skeletonization, each skeleton voxel is classified as *branch*, *line*, and *end* based on connectivity criteria as presented in section 4.4.5 (Fig. 4.32). From the fact that region classification can be done by classifying all the skeleton voxels, skeleton tree propagation is performed. As schematically illustrated in Fig. 4.33, *branch* and *end* voxels are iteratively classified. Starting from sub-roots that are to be propagated, all possible targets that are unclassified is pushed into the priority queue. The priority queue contains possible propagation candidates based on the directional consistency criterion (i.e., the similarity of the target direction with respect to the previous direction). A priority condition via directional consistency can be



(a)

(b)

Figure 4.34: Hepatic and portal vein reconstruction from two separate, classified skeletons.

formulated as

$$Priority\_value = \mathcal{B}_{c,i} = (||\mathbf{v}_{c,i} - \mathbf{v}_{c-1}||) \cdot (||\mathbf{v}_{c+1,j} - \mathbf{v}_{c,i}||), \tag{4.39}$$

where $\mathbf{v}$ indicates position of *branch* or *end* voxel with child index, c. i, j indicate indices for current (i.e., contained in the priority queue) and child, respectively. The priority is assigned by an anatomical knowledge that vessels are formed by smooth variations in angle. The propagation procedure is iteratively performed by pushing the unassigned candidates and poping the assigned. Figure 4.33 illustrates the proposed propagation algorithm. Finally, each vessel region is classified based on each related skeleton which is classified as either hepatic or portal vein (Fig. 4.34).

4.6 Discussion

The maximum intensity-based imaging is a very important technique compared to the single-thickness multi-planar reformation image from the perspective of clinical diagnosis and visualization [150, 173–176]. The maximum intensity technique can provide images of diagnostic quality as long as the contrast of the vessel of interest is sufficiently higher compared with that of surrounding structures. The proposed method showed that the projection can be particularly useful for depicting small vessels.

Segmentation of maximum intensity projection images can benefit the challenging task of fine vessel segmentation. However, a particular pixel in an image may arise from any voxel along the projection ray. If two or more vessels overlap in a certain direction, the vessel with a higher intensity is projected. The overlapping property makes it hard to reconstruct 3D structures via projection images especially with the complex liver vascular system. Inspired by sliding-thin-slab image analysis [150], strong vessel candidates in the 3D domain were successfully reconstructed based on maximum intensity slab images rather than 1-voxel thickness image or full-projected images. The proposed method performed the segmentation of multiple 2D slab images without any geometric assumptions (e.g., connectivity) and then back-projected points to the original 3D space to generate 3D vessel candidates. It was able to generate very dense vessel candidates that can aid the active contour model to extract accurate vessel region in the original image.

The proposed method finally extracted the smooth and fine structure of the complex 3D liver vascular system via a newly designed level set method. The model was designed by combining both region and local gradient energies with the help of vessel probability map which is generated by dense vessel candidates. The fine segments, (i.e., thin and weak peripheral branch vessels) whose boundaries are hard to be identified in the original CT image, were successfully segmented by strong and dense vessel candidates on minor regions. The experiments showed that the proposed model is superior to other models regarding the segmentation of small peripheral branch vessels without any manual interactions. Furthermore, the method presented a robust initialization metric that boosts the accuracy of active contour model approaches.

In pathological liver case (e.g., liver with the tumor), overall liver and tumor segmentation must be preceded to the proposed vessel segmentation. Tumor boundaries or tumor tissues might affect the vessel candidate generation step regarding vesselness filtering responses. False detection of vessel candidates may result in poor segmentation results. Therefore, for the effective use of the proposed algorithm, the maximum intensity slab image must be generated by excluding tumor regions.

Chapter 5

Conclusion and Future Works

The accurate segmentation of a liver and its vessels is still a challenging task. While deep learning continues to grow in influence until recently, the lack of annotated medical image data makes it difficult to successfully deploy CNNs in the clinics. Therefore, improving generalization performance is one of the most important element technologies for utilizing CNN. In this dissertation, a CNN for liver segmentation is proposed to minimize generalization errors based on the human-designed curriculum (i.e., auto-context). The proposed method minimized the error between train and test images more than other modern neural networks. In addition, the contour scheme has been successfully employed to the network by introducing a self-supervising approach. Instead of using the entire ground-truth contour, sparse contours have been trained so that the network can focus on its failures. Based on the experimental results, it was examined that the proposed method played a significant role in improving accuracy without introducing extra false positives.

CNN-based methods are difficult to be applied to vessel segmentation task because the annotation of a complex vascular structure is hard to be obtained. Therefore, an image-based segmentation algorithm is presented in the dissertation. To overcome the difficulty in thin vessel segmentation, a robust algorithm is proposed based on vessel candidate points obtained by using multiple maximum intensity projection images. Thin vessel branches (i.e., weak, peripheral vessels) were successfully segmented through vessel candidate points. In addition, an example application is presented to show that the portal vein can be easily separated from the hepatic vein when the contrast of vessels is well enhanced in portal phase CT images.

Further research is required to build a more intelligent and accurate computer-aided diagnosis system: 1) liver tumor segmentation, 2) liver partitioning using hepatic vein structure, and 3) image registration between multiple phases. The proposed liver and its vessel segmentation methods provide a basis for these algorithms. For example, the automated liver segmentation algorithm proposed in this dissertation can be used as a good prior knowledge in liver tumor segmentation task as similarly shown in vessel segmentation. Furthermore, the segmented hepatic vein can be anatomically categorized by branch, which allows the liver to be clearly partitioned. The partitioning of the liver region and the tumor segmentation together can make it possible to establish accurate surgical planning for liver resection. Finally, the proposed vessel segmentation algorithm presents the possibility of solving the registration problem between different phases. Since the proposed algorithm does not use the structural assumptions of the vessel (e.g., trees), it is possible to segment the contrast-enhanced vessel region in various phases. This suggests that the multi-phase registration problem, which is difficult to find matching points inside the liver parenchyma, can be solved by merging vessel structures. The automated phase-to-phase registration can be used for clearer clinical diagnosis in the future.

Bibliography

[1] B. Van Ginneken, C. M. Schaefer-Prokop, and M. Prokop, "Computer-aided diagnosis: how to move from the laboratory to the clinic," *Radiology*, vol. 261, no. 3, pp. 719–732, 2011.

[2] R. D. Howe and Y. Matsuoka, "Robotics for surgery," *Annual review of biomedical engineering*, vol. 1, no. 1, pp. 211–240, 1999.

[3] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[4] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.

[5] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," *arXiv preprint arXiv:1708.04552*, 2017.

[6] G. Cai, Y. Wang, L. He, and M. Zhou, "Unsupervised domain adaptation with adversarial residual transform networks," *IEEE Transactions on Neural Networks and Learning Systems*, 2019.

[7] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," *arXiv preprint arXiv:1502.02791*, 2015.

[8] G. Pereyra, G. Tucker, J. Chorowski, Ł. Kaiser, and G. Hinton, "Regularizing neural networks by penalizing confident output distributions," *arXiv preprint arXiv:1701.06548*, 2017.

[9] L. He, Z. Peng, B. Everding, X. Wang, C. Y. Han, K. L. Weiss, and W. G. Wee, "A comparative study of deformable contour methods on medical image segmentation," *Image and vision computing*, vol. 26, no. 2, pp. 141–163, 2008.

[10] H. Chen, X. Qi, L. Yu, Q. Dou, J. Qin, and P.-A. Heng, "Dcan: Deep contour-aware networks for object instance segmentation from histology images," *Medical image analysis*, vol. 36, pp. 135–146, 2017.

[11] Z. Tu and X. Bai, "Auto-context and its application to high-level vision tasks and 3d brain image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 10, pp. 1744–1757, 2010.

[12] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," California Univ San Diego La Jolla Inst for Cognitive Science, Tech. Rep., 1985.

[13] S. Haykin, *Neural networks: a comprehensive foundation.* Prentice Hall PTR, 1994.

[14] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[15] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[16] S. Jégou, M. Drozdzal, D. Vazquez, A. Romero, and Y. Bengio, “The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation,” in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on.* IEEE, 2017, pp. 1175–1183.

[17] B. Neyshabur, S. Bhojanapalli, D. McAllester, and N. Srebro, “Exploring generalization in deep learning,” in *Advances in Neural Information Processing Systems*, 2017, pp. 5947–5956.

[18] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” *arXiv preprint arXiv:1502.03167*, 2015.

[19] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 807–814.

[20] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[21] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[22] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu, “Deeply-supervised nets,” in *Artificial Intelligence and Statistics*, 2015, pp. 562–570.

[23] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.

[24] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[25] X. Li, L. Zhao, L. Wei, M.-H. Yang, F. Wu, Y. Zhuang, H. Ling, and J. Wang, "Deepsaliency: Multi-task deep neural network model for salient object detection," *IEEE Transactions on Image Processing*, vol. 25, no. 8, pp. 3919–3930, 2016.

[26] Q. Dou, L. Yu, H. Chen, Y. Jin, X. Yang, J. Qin, and P.-A. Heng, "3d deeply supervised network for automated segmentation of volumetric medical images," *Medical image analysis*, vol. 41, pp. 40–54, 2017.

[27] Z. Lin, M. Feng, C. N. d. Santos, M. Yu, B. Xiang, B. Zhou, and Y. Bengio, "A structured self-attentive sentence embedding," *arXiv preprint arXiv:1703.03130*, 2017.

[28] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[29] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *arXiv preprint arXiv:1508.04025*, 2015.

[30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[31] T. Shen, T. Zhou, G. Long, J. Jiang, S. Pan, and C. Zhang, "Disan: Directional self-attention network for rnn/cnn-free language understanding," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[32] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[33] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3156–3164.

[34] S. Jetley, N. A. Lord, N. Lee, and P. H. Torr, "Learn to pay attention," *arXiv preprint arXiv:1804.02391*, 2018.

[35] B. Zhao, J. Feng, X. Wu, and S. Yan, "A survey on deep learning-based fine-grained object classification and semantic segmentation," *International Journal of Automation and Computing*, vol. 14, no. 2, pp. 119–135, 2017.

[36] K. Li, Z. Wu, K.-C. Peng, J. Ernst, and Y. Fu, "Tell me where to look: Guided attention inference network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9215–9223.

[37] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille, "Attention to scale: Scale-aware semantic image segmentation," in *Proceedings of the*

IEEE conference on computer vision and pattern recognition, 2016, pp. 3640–3649.

[38] M. Ren and R. S. Zemel, "End-to-end instance segmentation with recurrent attention," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6656–6664.

[39] H. Li, P. Xiong, J. An, and L. Wang, "Pyramid attention network for semantic segmentation," *arXiv preprint arXiv:1805.10180*, 2018.

[40] H. Zhao, Y. Zhang, S. Liu, J. Shi, C. Change Loy, D. Lin, and J. Jia, "Psanet: Point-wise spatial attention network for scene parsing," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 267–283.

[41] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3146–3154.

[42] S. Chen, X. Tan, B. Wang, and X. Hu, "Reverse attention for salient object detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 234–250.

[43] J. Liu, G. Wang, P. Hu, L.-Y. Duan, and A. C. Kot, "Global context-aware attention lstm networks for 3d action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1647–1656.

[44] W. Pei, T. Baltrusaitis, D. M. Tax, and L.-P. Morency, "Temporal attention-gated model for robust sequence classification," in *Proceedings*

of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 6730–6739.

[45] X. Wang, R. Girshick, A. Gupta, and K. He, “Non-local neural networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7794–7803.

[46] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, “Show, attend and tell: Neural image caption generation with visual attention,” in *International conference on machine learning*, 2015, pp. 2048–2057.

[47] J. Lu, C. Xiong, D. Parikh, and R. Socher, “Knowing when to look: Adaptive attention via a visual sentinel for image captioning,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 375–383.

[48] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola, “Stacked attention networks for image question answering,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 21–29.

[49] H. Nam, J.-W. Ha, and J. Kim, “Dual attention networks for multimodal reasoning and matching,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 299–307.

[50] X. Chu, W. Yang, W. Ouyang, C. Ma, A. L. Yuille, and X. Wang, “Multi-context attention for human pose estimation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1831–1840.

[51] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[52] J. Schlemper, O. Oktay, M. Schaap, M. Heinrich, B. Kainz, B. Glocker, and D. Rueckert, "Attention gated networks: Learning to leverage salient regions in medical images," *Medical image analysis*, vol. 53, pp. 197–207, 2019.

[53] Y. Wang, Z. Deng, X. Hu, L. Zhu, X. Yang, X. Xu, P.-A. Heng, and D. Ni, "Deep attentional features for prostate segmentation in ultrasound," in *International Conference on Medical Image Computing and Computer-Assisted Intervention.* Springer, 2018, pp. 523–530.

[54] D. Nie, Y. Gao, L. Wang, and D. Shen, "Asdnet: Attention based semi-supervised deep networks for medical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention.* Springer, 2018, pp. 370–378.

[55] A. Sinha and J. Dolz, "Multi-scale guided attention for medical image segmentation," *arXiv preprint arXiv:1906.02849*, 2019.

[56] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[57] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 3–19.

[58] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3d u-net: learning dense volumetric segmentation from sparse anno-

tation," in *International conference on medical image computing and computer-assisted intervention.* Springer, 2016, pp. 424–432.

[59] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention.* Springer, 2015, pp. 234–241.

[60] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *3D Vision (3DV), 2016 Fourth International Conference on.* IEEE, 2016, pp. 565–571.

[61] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning.* MIT press, 2016.

[62] H. Chen, Q. Dou, L. Yu, J. Qin, and P.-A. Heng, "Voxresnet: Deep voxelwise residual networks for brain segmentation from 3d mr images," *NeuroImage*, 2017.

[63] E. Gibson, F. Giganti, Y. Hu, E. Bonmati, S. Bandula, K. Gurusamy, B. Davidson, S. P. Pereira, M. J. Clarkson, and D. C. Barratt, "Automatic multi-organ segmentation on abdominal ct with dense v-networks," *IEEE Transactions on Medical Imaging*, 2018.

[64] O. Oktay, J. Schlemper, L. L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N. Y. Hammerla, B. Kainz *et al.*, "Attention u-net: Learning where to look for the pancreas," *arXiv preprint arXiv:1804.03999*, 2018.

[65] H. R. Roth, H. Oda, Y. Hayashi, M. Oda, N. Shimizu, M. Fujiwara, K. Misawa, and K. Mori, “Hierarchical 3d fully convolutional networks for multi-organ segmentation,” *arXiv preprint arXiv:1704.06382*, 2017.

[66] H. R. Roth, L. Lu, N. Lay, A. P. Harrison, A. Farag, A. Sohn, and R. M. Summers, “Spatial aggregation of holistically-nested convolutional neural networks for automated pancreas localization and segmentation,” *Medical image analysis*, vol. 45, pp. 94–107, 2018.

[67] Q. Yu, L. Xie, Y. Wang, Y. Zhou, E. K. Fishman, and A. L. Yuille, “Recurrent saliency transformation network: Incorporating multi-stage visual cues for small organ segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8280–8289.

[68] M. Khened, V. A. Kollerathu, and G. Krishnamurthi, “Fully convolutional multi-scale residual densenets for cardiac segmentation and automated cardiac diagnosis using ensemble of classifiers,” *Medical image analysis*, vol. 51, pp. 21–45, 2019.

[69] D. Britz, A. Goldie, M.-T. Luong, and Q. Le, “Massive exploration of neural machine translation architectures,” *arXiv preprint arXiv:1703.03906*, 2017.

[70] P. Viola and M. J. Jones, “Robust real-time face detection,” *International journal of computer vision*, vol. 57, no. 2, pp. 137–154, 2004.

[71] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[72] Z. Tu, “Probabilistic boosting-tree: Learning discriminative models for classification, recognition, and clustering,” in *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, vol. 2. IEEE, 2005, pp. 1589–1596.

[73] S. S. M. Salehi, D. Erdogmus, and A. Gholipour, “Auto-context convolutional neural network (auto-net) for brain extraction in magnetic resonance imaging,” *IEEE transactions on medical imaging*, vol. 36, no. 11, pp. 2319–2330, 2017.

[74] R. Gan, W. C. Wong, and A. C. Chung, “Statistical cerebrovascular segmentation in three-dimensional rotational angiography based on maximum intensity projections,” *Medical physics*, vol. 32, no. 9, pp. 3017–3028, 2005.

[75] B. E. Chapman, J. O. Stapelton, and D. L. Parker, “Intracranial vessel segmentation from time-of-flight mra using pre-processing of the mip z-buffer: accuracy of the zbs algorithm,” *Medical Image Analysis*, vol. 8, no. 2, pp. 113–126, 2004.

[76] D. L. Parker, B. E. Chapman, J. A. Roberts, A. L. Alexander, and J. S. Tsuruda, “Enhanced image detail using continuity in the mip z-buffer: Applications to magnetic resonance angiography,” *Journal of Magnetic Resonance Imaging*, vol. 11, no. 4, pp. 378–388, 2000.

[77] S.-J. Lim, Y.-Y. Jeong, and Y.-S. Ho, “Automatic liver segmentation for volume measurement in ct images,” *Journal of Visual Communication and Image Representation*, vol. 17, no. 4, pp. 860–875, 2006.

[78] L. Rusko, G. Bekes, G. Nemeth, and M. Fidrich, “Fully automatic liver segmentation for contrast-enhanced ct images,” *MICCAI Wshp. 3D Segmentation in the Clinic: A Grand Challenge*, vol. 2, no. 7, 2007.

[79] K. Suzuki, R. Kohlbrenner, M. L. Epstein, A. M. Obajuluwa, J. Xu, and M. Hori, “Computer-aided measurement of liver volumes in ct by means of geodesic active contour segmentation coupled with level-set algorithms,” *Medical physics*, vol. 37, no. 5, pp. 2159–2166, 2010.

[80] J. Lee, N. Kim, H. Lee, J. B. Seo, H. J. Won, Y. M. Shin, Y. G. Shin, and S.-H. Kim, “Efficient liver segmentation using a level-set method with optimal detection of the initial liver boundary from level-set speed images,” *Computer Methods and Programs in Biomedicine*, vol. 88, no. 1, pp. 26–38, 2007.

[81] G. Li, X. Chen, F. Shi, W. Zhu, J. Tian, and D. Xiang, “Automatic liver segmentation based on shape constraints and deformable graph cut in ct images,” *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5315–5329, 2015.

[82] T. Heimann, B. Van Ginneken, M. A. Styner, Y. Arzhaeva, V. Aurich, C. Bauer, A. Beck, C. Becker, R. Beichel, G. Bekes *et al.*, “Comparison and evaluation of methods for liver segmentation from ct datasets,” *IEEE transactions on medical imaging*, vol. 28, no. 8, pp. 1251–1265, 2009.

[83] Q. Huang, H. Ding, X. Wang, and G. Wang, “Fully automatic liver segmentation in ct images using modified graph cuts and feature detection,” *Computers in biology and medicine*, vol. 95, pp. 198–208, 2018.

[84] S. Zheng, B. Fang, L. Li, M. Gao, and Y. Wang, “A variational approach to liver segmentation using statistics from multiple sources,” *Physics in Medicine & Biology*, vol. 63, no. 2, p. 025024, 2018.

[85] Y. Yuan, Y.-W. Chen, C. Dong, H. Yu, and Z. Zhu, “Hybrid method combining superpixel, random walk and active contour model for fast and accurate liver segmentation,” *Computerized Medical Imaging and Graphics*, vol. 70, pp. 119–134, 2018.

[86] X. Guo, L. H. Schwartz, and B. Zhao, “Automatic liver segmentation by integrating fully convolutional networks into active contour models,” *Medical physics*, 2019.

[87] W. Qin, J. Wu, F. Han, Y. Yuan, W. Zhao, B. Ibragimov, J. Gu, and L. Xing, “Superpixel-based and boundary-sensitive convolutional neural network for automated liver segmentation,” *Physics in Medicine & Biology*, vol. 63, no. 9, p. 095017, 2018.

[88] T. Heimann and H.-P. Meinzer, “Statistical shape models for 3d medical image segmentation: a review,” *Medical image analysis*, vol. 13, no. 4, pp. 543–563, 2009.

[89] X. Wang, Y. Zheng, L. Gan, X. Wang, X. Sang, X. Kong, and J. Zhao, “Liver segmentation from ct images using a sparse priori statistical shape model (sp-ssm),” *PloS one*, vol. 12, no. 10, p. e0185249, 2017.

[90] A. Hoover, V. Kouznetsova, and M. Goldbaum, “Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response,” *IEEE Transactions on Medical imaging*, vol. 19, no. 3, pp. 203–210, 2000.

[91] X. Jiang and D. Mojon, “Adaptive local thresholding by verification-based multithreshold probing with application to vessel detection in retinal images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 1, pp. 131–137, 2003.

[92] B. Bouraoui, C. Ronse, J. Baruthio, N. Passat, and P. Germain, “3d segmentation of coronary arteries based on advanced mathematical morphology techniques,” *Computerized medical imaging and graphics*, vol. 34, no. 5, pp. 377–387, 2010.

[93] A. M. Mendonca and A. Campilho, “Segmentation of retinal blood vessels by combining the detection of centerlines and morphological reconstruction,” *IEEE transactions on medical imaging*, vol. 25, no. 9, pp. 1200–1213, 2006.

[94] F. Rossant, M. Badellino, A. Chavillon, I. Bloch, and M. Paques, “A morphological approach for vessel segmentation in eye fundus images, with quantitative evaluation,” *Journal of Medical Imaging and Health Informatics*, vol. 1, no. 1, pp. 42–49, 2011.

[95] F. Zana and J.-C. Klein, “Segmentation of vessel-like patterns using mathematical morphology and curvature evaluation,” *IEEE transactions on image processing*, vol. 10, no. 7, pp. 1010–1019, 2001.

[96] C. Kirbas and F. K. Quek, “Vessel extraction in medical images by 3d wave propagation and traceback,” in *Third IEEE Symposium on Bioinformatics and Bioengineering, 2003. Proceedings.* IEEE, 2003, pp. 174–181.

[97] M. E. Martínez-Pérez, A. D. Hughes, A. V. Stanton, S. A. Thom, A. A. Bharath, and K. H. Parker, “Retinal blood vessel segmentation by means

of scale-space analysis and region growing," in *International Conference on Medical Image Computing and Computer-Assisted Intervention.* Springer, 1999, pp. 90–97.

[98] C. Metz, M. Schaap, A. Van Der Giessen, T. Van Walsum, and W. Niessen, "Semi-automatic coronary artery centerline extraction in computed tomography angiography data," in *2007 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro.* IEEE, 2007, pp. 856–859.

[99] F. K. Quek and C. Kirbas, "Vessel extraction in medical images by wave-propagation and traceback," *IEEE transactions on Medical Imaging*, vol. 20, no. 2, pp. 117–131, 2001.

[100] C. Revol-Muller, F. Peyrin, Y. Carrillon, and C. Odet, "Automated 3d region growing algorithm based on an assessment function," *Pattern Recognition Letters*, vol. 23, no. 1-3, pp. 137–150, 2002.

[101] J. Yi and J. B. Ra, "A locally adaptive region growing algorithm for vascular segmentation," *International Journal of Imaging Systems and Technology*, vol. 13, no. 4, pp. 208–214, 2003.

[102] A. F. Frangi, W. J. Niessen, K. L. Vincken, and M. A. Viergever, "Multiscale vessel enhancement filtering," in *International conference on medical image computing and computer-assisted intervention.* Springer, 1998, pp. 130–137.

[103] S. Chaudhuri, S. Chatterjee, N. Katz, M. Nelson, and M. Goldbaum, "Detection of blood vessels in retinal images using two-dimensional matched filters," *IEEE Transactions on medical imaging*, vol. 8, no. 3, pp. 263–269, 1989.

[104] G. Läthén, J. Jonasson, and M. Borga, "Blood vessel segmentation using multi-scale quadrature filtering," *Pattern Recognition Letters*, vol. 31, no. 8, pp. 762–767, 2010.

[105] R. Manniesing, M. A. Viergever, and W. J. Niessen, "Vessel enhancing diffusion: A scale space representation of vessel structures," *Medical image analysis*, vol. 10, no. 6, pp. 815–825, 2006.

[106] Y. Sato, S. Nakajima, H. Atsumi, T. Koller, G. Gerig, S. Yoshida, and R. Kikinis, "3d multi-scale line filter for segmentation and visualization of curvilinear structures in medical images," in *CVRMed-MRCAS'97*. Springer, 1997, pp. 213–222.

[107] P. T. Truc, M. A. Khan, Y.-K. Lee, S. Lee, and T.-S. Kim, "Vessel enhancement filter using directional filter bank," *Computer Vision and Image Understanding*, vol. 113, no. 1, pp. 101–112, 2009.

[108] A. A.-H. A.-R. Youssif, A. Z. Ghalwash, and A. A. S. A.-R. Ghoneim, "Optic disc detection from normalized digital fundus images by means of a vessels' direction matched filter," *IEEE transactions on medical imaging*, vol. 27, no. 1, pp. 11–18, 2007.

[109] Y. Sato, S. Nakajima, N. Shiraga, H. Atsumi, S. Yoshida, T. Koller, G. Gerig, and R. Kikinis, "Three-dimensional multi-scale line filter for segmentation and visualization of curvilinear structures in medical images," *Medical image analysis*, vol. 2, no. 2, pp. 143–168, 1998.

[110] T. M. Koller, G. Gerig, G. Szekely, and D. Dettwiler, "Multiscale detection of curvilinear structures in 2-d and 3-d image data," in *Proceedings of IEEE International Conference on Computer Vision*. IEEE, 1995, pp. 864–869.

[111] N. Flasque, M. Desvignes, J.-M. Constans, and M. Revenu, “Acquisition, segmentation and tracking of the cerebral vascular tree on 3d magnetic resonance angiography images,” *Medical Image Analysis*, vol. 5, no. 3, pp. 173–183, 2001.

[112] O. Friman, M. Hindennach, C. Kühnel, and H.-O. Peitgen, “Multiple hypothesis template tracking of small 3d vessel structures,” *Medical image analysis*, vol. 14, no. 2, pp. 160–171, 2010.

[113] Y. Sun, “Automated identification of vessel contours in coronary arteriograms by an adaptive tracking algorithm,” *IEEE transactions on medical imaging*, vol. 8, no. 1, pp. 78–88, 1989.

[114] M. Vlachos and E. Dermatas, “Multi-scale retinal vessel segmentation using line tracking,” *Computerized Medical Imaging and Graphics*, vol. 34, no. 3, pp. 213–227, 2010.

[115] O. Wink, W. J. Niessen, and M. A. Viergever, “Multiscale vessel tracking,” *IEEE Transactions on Medical Imaging*, vol. 23, no. 1, pp. 130–133, 2004.

[116] I. Liu and Y. Sun, “Recursive tracking of vascular networks in angiograms based on the detection-deletion scheme,” *IEEE Transactions on medical imaging*, vol. 12, no. 2, pp. 334–341, 1993.

[117] U. T. Nguyen, A. Bhuiyan, L. A. Park, and K. Ramamohanarao, “An effective retinal blood vessel segmentation method using multi-scale line detection,” *Pattern recognition*, vol. 46, no. 3, pp. 703–715, 2013.

[118] S. R. Aylward and E. Bullitt, “Initialization, noise, singularities, and scale in height ridge traversal for tubular object centerline extraction,” *IEEE transactions on medical imaging*, vol. 21, no. 2, pp. 61–75, 2002.

[119] J. Staal, M. D. Abràmoff, M. Niemeijer, M. A. Viergever, and B. Van Ginneken, “Ridge-based vessel segmentation in color images of the retina,” *IEEE transactions on medical imaging*, vol. 23, no. 4, pp. 501–509, 2004.

[120] C. Kirbas and F. Quek, “A review of vessel extraction techniques and algorithms,” *ACM Computing Surveys (CSUR)*, vol. 36, no. 2, pp. 81–121, 2004.

[121] J. F. Barrett and N. Keat, “Artifacts in ct: recognition and avoidance,” *Radiographics*, vol. 24, no. 6, pp. 1679–1691, 2004.

[122] M. Kass, A. Witkin, and D. Terzopoulos, “Snakes: Active contour models,” *International journal of computer vision*, vol. 1, no. 4, pp. 321–331, 1988.

[123] S. Lankton and A. Tannenbaum, “Localizing region-based active contours,” *IEEE transactions on image processing*, vol. 17, no. 11, pp. 2029–2039, 2008.

[124] C. Li, C. Xu, C. Gui, and M. D. Fox, “Level set evolution without re-initialization: a new variational formulation,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, vol. 1. IEEE, 2005, pp. 430–436.

[125] ——, “Distance regularized level set evolution and its application to image segmentation,” *IEEE transactions on image processing*, vol. 19, no. 12, pp. 3243–3254, 2010.

[126] J. A. Sethian, "Evolution, implementation, and application of level set and fast marching methods for advancing fronts," *Journal of computational physics*, vol. 169, no. 2, pp. 503–555, 2001.

[127] C. Li, C.-Y. Kao, J. C. Gore, and Z. Ding, "Implicit active contours driven by local binary fitting energy," in *2007 IEEE Conference on Computer Vision and Pattern Recognition.* IEEE, 2007, pp. 1–7.

[128] ——, "Minimization of region-scalable fitting energy for image segmentation," *IEEE transactions on image processing*, vol. 17, no. 10, pp. 1940–1949, 2008.

[129] J. A. Sethian, "A fast marching level set method for monotonically advancing fronts," *Proceedings of the National Academy of Sciences*, vol. 93, no. 4, pp. 1591–1595, 1996.

[130] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," in *Proceedings of IEEE international conference on computer vision.* IEEE, 1995, pp. 694–699.

[131] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Transactions on image processing*, vol. 10, no. 2, pp. 266–277, 2001.

[132] H. Song, Y. Zheng, and K. Zhang, "Efficient algorithm for piecewise-smooth model with approximately explicit solutions," *Electronics Letters*, vol. 53, no. 4, pp. 233–235, 2017.

[133] K. Zhang, L. Zhang, K.-M. Lam, and D. Zhang, "A level set approach to image segmentation with intensity inhomogeneity," *IEEE transactions on cybernetics*, vol. 46, no. 2, pp. 546–557, 2015.

[134] L. M. Lorigo, O. D. Faugeras, W. E. L. Grimson, R. Keriven, R. Kikinis, A. Nabavi, and C.-F. Westin, “Curves: Curve evolution for vessel segmentation,” *Medical image analysis*, vol. 5, no. 3, pp. 195–206, 2001.

[135] Y. Shang, R. Deklerck, E. Nyssen, A. Markova, J. de Mey, X. Yang, and K. Sun, “Vascular active contour for vessel tree segmentation,” *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 4, pp. 1023–1032, 2010.

[136] K. Sun, Z. Chen, and S. Jiang, “Local morphology fitting active contour for automatic vascular segmentation,” *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 2, pp. 464–473, 2011.

[137] R. Toledo, X. Orriols, P. Radeva, X. Binefa, J. Vitria, C. Canero, and J. Villanuev, “Eigensnakes for vessel segmentation in angiography,” in *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, vol. 4. IEEE, 2000, pp. 340–343.

[138] Y. Zhao, L. Rada, K. Chen, S. P. Harding, and Y. Zheng, “Automated vessel segmentation using infinite perimeter active contour model with hybrid region information with application to retinal images,” *IEEE transactions on medical imaging*, vol. 34, no. 9, pp. 1797–1807, 2015.

[139] S. Osher and J. A. Sethian, “Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations,” *Journal of computational physics*, vol. 79, no. 1, pp. 12–49, 1988.

[140] D. Mumford and J. Shah, “Optimal approximations by piecewise smooth functions and associated variational problems,” *Communications on pure and applied mathematics*, vol. 42, no. 5, pp. 577–685, 1989.

[141] P. Yan and A. A. Kassim, “Segmentation of volumetric mra images by using capillary active contour,” *Medical Image Analysis*, vol. 10, no. 3, pp. 317–329, 2006.

[142] R. Manniesing, B. K. Velthuis, M. S. van Leeuwen, I. Van Der Schaaf, P. Van Laar, and W. J. Niessen, “Level set based cerebral vasculature segmentation and diameter quantification in ct angiography,” *Medical image analysis*, vol. 10, no. 2, pp. 200–214, 2006.

[143] S. C. Zhu and A. Yuille, “Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation,” *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 9, pp. 884–900, 1996.

[144] G. B. Arfken and H. J. Weber, “Mathematical methods for physicists,” 1999.

[145] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[146] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256.

[147] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[148] D.-J. Kroon, "Smooth triangulated mesh," File Exchange - MATLAB Central, 2010 (accessed: Jun. 8, 2018). [Online]. Available: http://mathworks.com/matlabcentral/fileexchange/26710

[149] K. Suzuki, I. Horiba, and N. Sugie, "Linear-time connected-component labeling based on sequential local operations," *Computer Vision and Image Understanding*, vol. 89, no. 1, pp. 1–23, 2003.

[150] B. Ertl-Wagner, R. Bruening, J. Blume, R.-T. Hoffmann, S. Mueller-Schunk, B. Snyder, and M. Reiser, "Relative value of sliding-thin-slab multiplanar reformations and sliding-thin-slab maximum intensity projections as reformatting techniques in multisection ct angiography of the cervicocranial vessels," *American journal of neuroradiology*, vol. 27, no. 1, pp. 107–113, 2006.

[151] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE Transactions on image processing*, vol. 16, no. 8, pp. 2080–2095, 2007.

[152] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images." in *Iccv*, vol. 98, no. 1, 1998, p. 2.

[153] Q. Yang, K.-H. Tan, and N. Ahuja, "Real-time o (1) bilateral filtering," in *2009 IEEE Conference on Computer Vision and Pattern Recognition.* IEEE, 2009, pp. 557–564.

[154] Y. Yu and S. T. Acton, "Speckle reducing anisotropic diffusion," *IEEE Transactions on image processing*, vol. 11, no. 11, pp. 1260–1270, 2002.

[155] A. Beck and M. Teboulle, "Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems," *IEEE transactions on image processing*, vol. 18, no. 11, pp. 2419–2434, 2009.

[156] A. Buades, B. Coll, and J.-M. Morel, "Non-local means denoising," *Image Processing On Line*, vol. 1, pp. 208–212, 2011.

[157] P. Coupé, P. Yger, S. Prima, P. Hellier, C. Kervrann, and C. Barillot, "An optimized blockwise nonlocal means denoising filter for 3-d magnetic resonance images," *IEEE transactions on medical imaging*, vol. 27, no. 4, pp. 425–441, 2008.

[158] A. Buades, B. Coll, and J.-M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Modeling & Simulation*, vol. 4, no. 2, pp. 490–530, 2005.

[159] C. Kervrann and J. Boulanger, "Optimal spatial adaptation for patch-based image denoising," *IEEE Transactions on Image Processing*, vol. 15, no. 10, pp. 2866–2878, 2006.

[160] L. Sendur and I. W. Selesnick, "Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency," *IEEE Transactions on signal processing*, vol. 50, no. 11, pp. 2744–2756, 2002.

[161] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of gaussians in the wavelet domain," *IEEE Trans Image Processing*, vol. 12, no. 11, 2003.

[162] L. M. Florack, B. M. ter Haar Romeny, J. J. Koenderink, and M. A. Viergever, "Scale and the differential structure of images," *Image and vision computing*, vol. 10, no. 6, pp. 376–388, 1992.

[163] J. J. Koenderink, “The structure of images,” *Biological cybernetics*, vol. 50, no. 5, pp. 363–370, 1984.

[164] N. Otsu, “A threshold selection method from gray-level histograms,” *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[165] P.-O. Persson, “The level set method,” *Lecture Notes, MIT*, vol. 16, 2005.

[166] C. Arcelli, G. S. di Baja, and L. Serino, “Distance-driven skeletonization in voxel images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 4, pp. 709–720, 2010.

[167] I. Bitter, A. E. Kaufman, and M. Sato, “Penalized-distance volumetric skeleton algorithm,” *IEEE Transactions on Visualization and computer Graphics*, vol. 7, no. 3, pp. 195–206, 2001.

[168] W. Xie, R. P. Thompson, and R. Perucchio, “A topology-preserving parallel 3d thinning algorithm for extracting the curve skeleton,” *Pattern Recognition*, vol. 36, no. 7, pp. 1529–1544, 2003.

[169] 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.

[170] S. Lobregt, P. W. Verbeek, and F. C. Groen, “Three-dimensional skeletonization: principle and algorithm,” *IEEE Transactions on pattern analysis and machine intelligence*, no. 1, pp. 75–77, 1980.

[171] C. Pudney, “Distance-ordered homotopic thinning: a skeletonization algorithm for 3d digital images,” *Computer Vision and Image Understanding*, vol. 72, no. 3, pp. 404–413, 1998.

[172] P.-C. Liu, F.-C. Wu, W.-C. Ma, R.-H. Liang, and M. Ouhyoung, "Automatic animation skeleton using repulsive force field," in *11th Pacific Conference onComputer Graphics and Applications, 2003. Proceedings.* IEEE, 2003, pp. 409–413.

[173] S. Diederich, M. Lentschig, T. Overbeck, D. Wormanns, and W. Heindel, "Detection of pulmonary nodules at spiral ct: comparison of maximum intensity projection sliding slabs and single-image reporting," *European radiology*, vol. 11, no. 8, pp. 1345–1350, 2001.

[174] J. F. Gruden, S. Ouanounou, S. Tigges, S. D. Norris, and T. S. Klausner, "Incremental benefit of maximum-intensity-projection images on observer detection of small pulmonary nodules revealed by multidetector ct," *American Journal of Roentgenology*, vol. 179, no. 1, pp. 149–157, 2002.

[175] N. Kawel, B. Seifert, M. Luetolf, and T. Boehm, "Effect of slab thickness on the ct detection of pulmonary nodules: use of sliding thin-slab maximum intensity projection and volume rendering," *American Journal of Roentgenology*, vol. 192, no. 5, pp. 1324–1329, 2009.

[176] M. Prokop, H. O. Shin, A. Schanz, and C. M. Schaefer-Prokop, "Use of maximum intensity projections in ct angiography: a basic review." *Radiographics*, vol. 17, no. 2, pp. 433–451, 1997.

초록

복부 전산화 단층 촬영 (CT) 영상에서 정확한 간 및 혈관 분할은 체적 측정, 치료 계획 수립 및 추가적인 증강 현실 기반 수술 가이드와 같은 컴퓨터 진단 보조 시스템을 구축하는데 필수적인 요소이다. 최근 들어 컨볼루셔널 인공 신경망 (CNN) 형태의 딥 러닝이 많이 적용되면서 의료 영상 분할의 성능이 향상되고 있지만, 실제 임상에 적용할 수 있는 높은 일반화 성능을 제공하기는 여전히 어렵다. 또한 물체의 경계는 전통적으로 영상 분할에서 매우 중요한 요소로 이용되었지만, CT 영상에서 간의 불분명한 경계를 추출하기가 어렵기 때문에 현대 CNN에서는 이를 사용하지 않고 있다. 간 혈관 분할 작업의 경우, 복잡한 혈관 영상으로부터 학습 데이터를 만들기 어렵기 때문에 딥 러닝을 적용하기가 어렵다. 또한 얇은 혈관 부분의 영상 밝기 대비가 약하여 원본 영상에서 식별하기가 매우 어렵다. 본 논문에서는 위 언급한 문제들을 해결하기 위해 일반화 성능이 향상된 CNN과 얇은 혈관을 포함하는 복잡한 간 혈관을 정확하게 분할하는 알고리즘을 제안한다.

간 분할 작업에서 우수한 일반화 성능을 갖는 CNN을 구축하기 위해, 내부적으로 간 모양을 추정하는 부분이 포함된 자동 컨텍스트 알고리즘을 제안한다. 또한, CNN을 사용한 학습에 경계선의 개념이 새롭게 제안된다. 모호한 경계부가 포함되어 있어 전체 경계 영역을 CNN에 훈련하는 것은 매우 어렵기 때문에 반복되는 학습 과정에서 인공 신경망이 스스로 예측한 확률에서 부정확하게 추정된 부분적 경계만을 사용하여 인공 신경망을 학습한다. 실험적 결과를 통해 제안된 CNN이 다른 최신 기법들보다 정확도가 우수하다는 것을 보인다. 또한, 제안된 CNN의 일반화 성능을 검증하기 위해 다양한 실험을 수행한다.

간 혈관 분할에서는 간 내부의 관심 영역을 지정하기 위해 앞서 획득한 간 영역을 활용한다. 정확한 간 혈관 분할을 위해 혈관 후보 점들을 추출하여 사용하는 알고리즘을 제안한다. 확실한 후보 점들을 얻기 위해, 삼차원 영상의 차원을 먼저

최대 강도 투영 기법을 통해 이차원으로 낮춘다. 이차원 영상에서는 복잡한 혈관의 구조가 보다 단순화될 수 있다. 이어서, 이차원 영상에서 혈관 분할을 수행하고 혈관 픽셀들은 원래의 삼차원 공간상으로 역 투영된다. 마지막으로, 전체 혈관의 분할을 위해 원본 영상과 혈관 후보 점들을 모두 사용하는 새로운 레벨 셋 기반 알고리즘을 제안한다. 제안된 알고리즘은 복잡한 구조가 단순화되고 얇은 혈관이 더 잘 보이는 이차원 영상에서 얻은 후보 점들을 사용하기 때문에 얇은 혈관 분할에서 높은 정확도를 보인다. 실험적 결과에 의하면 제안된 알고리즘은 잘못된 영역의 추출 없이 다른 레벨 셋 기반 알고리즘들보다 우수한 성능을 보인다.

제안된 알고리즘은 간과 혈관을 분할하는 새로운 방법을 제시한다. 제안된 자동 컨텍스트 구조는 사람이 디자인한 학습 과정이 일반화 성능을 크게 향상할 수 있다는 것을 보인다. 그리고 제안된 경계선 학습 기법으로 CNN을 사용한 영상 분할의 성능을 향상할 수 있음을 내포한다. 간 혈관의 분할은 이차원 최대 강도 투영 기반 이미지로부터 획득된 혈관 후보 점들을 통해 얇은 혈관들이 성공적으로 분할될 수 있음을 보인다. 본 논문에서 제안된 알고리즘은 간의 해부학적 분석과 자동화된 컴퓨터 진단 보조 시스템을 구축하는 데 매우 중요한 기술이다.

주요어: 간 분할, 자동 컨텍스트 인공 신경망, 윤곽선 집중 학습 기법, 혈관 후보 점, 혈관 분할, 레벨 셋 모델

학번: 2014-21778