INTERACTIVE MEDICAL IMAGE SEGMENTATION USING DEEP LEARNING WITH IMAGE SPECIFIC FINE TUNING

Seminar Report

Submitted in partial fulfillment of the requirements for
the award of degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

of

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

Submitted By

RESHMA REGUNATH

Department of Computer Science & Engineering
Mar Athanasius College Of Engineering Kothamangalam

INTERACTIVE MEDICAL IMAGE SEGMENTATION USING DEEP LEARNING WITH IMAGE SPECIFIC FINE TUNING

Seminar Report

Submitted in partial fulfillment of the requirements for the award of degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

of

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

Submitted By

RESHMA REGUNATH

Department of Computer Science & Engineering
Mar Athanasius College Of Engineering Kothamangalam

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MAR ATHANASIUS COLLEGE OF ENGINEERING
KOTHAMANGALAM

CERTIFICATE

This is to certify that the report entitled **Interactive Medical Image Segmentation Using Deep Learning with Image Specific Fine Tuning** *submitted by* **Ms. RESHMA REGHUNATH, Reg. No.MAC15CS048** *towards partial fulfillment of the requirement for the award of Degree of Bachelor of Technology in Computer science and Engineering from APJ Abdul Kalam Technological University for December 2018 is a bonafide record of the seminar carried out by her under our supervision and guidance.*

....	....	....
Prof. Joby George	**Prof. Neethu Subash**	**Dr. Surekha Mariam Varghese**
Faculty Guide	***Faculty Guide***	***Head Of Department***

Date: Dept. Seal

ACKNOWLEDGEMENT

First and foremost, I sincerely thank the 'God Almighty' for his grace for the successful and timely completion of the seminar.

I express my sincere gratitude and thanks to Dr. Solly George, Principal and Dr. Surekha Mariam Varghese, Head Of the Department for providing the necessary facilities and their encouragement and support.

I owe special thanks to the staff-in-charge Prof. Joby george, Prof. Neethu Subash and Prof. Joby Anu Mathew for their corrections, suggestions and sincere efforts to co-ordinate the seminar under a tight schedule.

I express my sincere thanks to staff members in the Department of Computer Science and Engineering who have taken sincere efforts in helping me to conduct this seminar.

Finally, I would like to acknowledge the heartfelt efforts, comments, criticisms, co-operation and tremendous support given to me by my dear friends during the preparation of the seminar and also during the presentation without whose support this work would have been all the more difficult to accomplish.

ABSTRACT

Convolutional neural network have achieved state-of-the-art performance for automatic medical image segmentation. To add image-specific adaptation and generalizability to previously unseen object classes a novel deep learning based interactive segmentation is introduced. The framework incorporate CNNs into a bounding box and scribble based segmentation pipeline. Bounding box extract foreground from a given region of interest. Weighted loss function considering network and interaction based uncertainty for the fine tuning is used. 2D segmentation of fetal placenta and 3D segmentation of brain tumor core for training is employed. Bounding box in test images are provided by users or through automatic detection further increases the efficiency. This method leads to accurate results with fewer user interactions and less user time than traditional interactive segmentation method.

Contents

List of Figures

LIST OF ABBREVIATION

CNN	Convolutional neural network
MRI	Magnetic Resonance Image
BIF	Bounding box and Image-specific Fine-tuning
GMM	Gaussian Mixture Model
CRF	Conditional Random Field
GPU	Graphical Processing Unit
SSFSE	Single-shot Fast Spin Echo
BRATS	Brain Tumor Segmentation Challenge
FLAIR	fluid attenuated inversion recovery
FCN	Fully Convolutional Network
CT	Computed Tomography

Introduction

DEEP learning with convolutional neural networks (CNNs) has achieved state-of-the-art performance for automated medical image segmentation. However, automatic segmentation methods have not demonstrated sufficiently accurate and robust results for clinical use due to the inherent challenges of medical images, such as poor image quality, different imaging and segmentation protocols, and variations among patients [2]. Alternatively, interactive segmentation methods are widely adopted, as they integrate the user's knowledge and take into account the application requirements for more robust segmentation performance. As such, interactive segmentation remains the state of the art for existing commercial surgical planning and navigation products. Though leveraging user interactions often leads to more robust segmentations, an interactive method should require as short user time as possible to reduce the burden on users. Motivated by these observations, we investigate combining CNNs with user interactions for medical image segmentation to achieve higher segmentation accuracy and robustness with fewer user interactions and less user time. However, there are very few studies on using CNNs for interactive segmentation. This is mainly due to the requirement of large amounts of annotated images for training, the lack of imagespecific adaptation and the demanding balance among model complexity, inference time and memory space efficiency.

The first challenge of using CNNs for interactive segmentation is that current CNNs do not generalize well to previously unseen object classes that are not present in the training set. As a result, they require labeled instances of each object class to be present in the training set. annotations are often expensive to acquire as both expertise and time are needed to produce accurate annotations. This limits the performance of CNNs to segment objects for which annotations are not available in the training stage.

Second, interactive segmentation often requires imagespecific learning to deal with large context variations among different images, but current CNNs are not adaptive to different test images, as parameters of the model are learned from training images and then fixed in the testing stage, without image-specific adaptation. It has been shown that image-specific adaptation of a pre-trained Gaussian Mixture Model (GMM) helps to improve segmentation accuracy. However, transitioning from simple GMMs to powerful but complex CNNs in this context has not yet been demonstrated.

Third, fast inference and memory efficiency are demanded for interactive segmentation. They can be relatively easily achieved for 2D images, but become much more problematic for 3D images. For example, DeepMedic [2] works on 3D local patches to reduce memory requirements but results in a slow inference. HighRes3DNet works on 3D whole images with relatively fast inference but needs a large amount of GPU memory, leading to high hardware requirements. To make a CNN-based interactive segmentation method efficient to use, enabling CNNs to respond quickly to user interactions and to work on a machine with limited GPU resources (e.g., a standard desktop PC or a laptop) is desirable. DeepIGeoS combines CNNs with user interactions and has demonstrated good interactivity. However, it has a lack of adaptability to unseen image contexts.

This paper presents a new framework to address these challenges for deep learning-based interactive segmentation. To generalize to previously unseen objects, we propose a bounding-box-based segmentation pipeline that extracts the foreground from a given region of interest, and design a 2D and a 3D CNN with good compactness to avoid over-fitting. To make CNNs adaptive to different test images, we propose image-specific fine-tuning. In addition, our networks consider a balance among receptive field, inference time and memory efficiency so as to be responsive to user interactions and have low requirements in terms of GPU resources.

Contributions

The contributions of this work are four-fold. First, we propose a novel deep learning-based framework for interactive 2D and 3D medical image segmentation by incorporating CNNs into a bounding box and scribble-based binary segmentation pipeline. Second, we propose image-specific fine-tuning to adapt a CNN model to each test image independently. The fine-tuning can be either unsupervised (without additional user interactions) or supervised by user-provided scribbles. Third, we propose a weighted loss function considering network and interaction-based uncertainty during the image-specific fine-tuning. Fourth, we present the first attempt to employ CNNs to deal with previously unseen objects (a.k.a. zeroshot learning) in the context of image segmentation. The proposed framework does not require all the object classes to be annotated for training. Thus, it can be applied to new organs or new segmentation protocols directly.

Related Works

2.1 CNNs for Image Segmentation

For natural image segmentation, FCN and DeepLab are among the state-of-the-art performing methods. For 2D biomedical image segmentation, efficient networks such as U-Net, DCAN and Nabla-net have been proposed. For 3D volumes, patch-based CNNs have been proposed for segmentation of the brain tumor and pancreas, and more powerful end-to-end 3D CNNs include V-Net [3], HighRes3DNet , and 3D deeply supervised network .

2.2 Interactive Segmentation Methods

A wide range of interactive segmentation methods have been proposed. Representative methods include Graph Cuts , Random Walks and GeoS . Machine learning has been popularly used to achieve high accuracy and interaction efficiency. For example, GMMs are used by GrabCut to segment color images. Online Random Forests (ORFs) are employed by Slic-Seg for placenta segmentation from fetal Magnetic Resonance images (MRI). Active learning is used to segment 3D Computed Tomography (CT) images. They have achieved more accurate segmentations with fewer user interactions than traditional interactive segmentation methods.

To combine user interactions with CNNs, DeepCut and ScribbleSup [4] propose to leverage user-provided bounding boxes or scribbles, but they employ user interactions as sparse annotations for the training set rather than as guidance for dealing with test images. 3D U-Net learns from annotations of some slices in a volume and produces a dense 3D segmentation, but is not responsive to user interactions. An FCN is combined with user interactions for 2D RGB image segmentation, without adaptation for medical images. DeepIGeoS uses geodesic distance transforms of scribbles as additional channels of CNNs for interactive segmentation, but cannot deal with previously unseen object classes.

2.3 Model Adaptation

Previous learning-based interactive segmentation methods often employ image-specific models. For example, GrabCut and Slic-Seg learn from the target image with GMMs and ORFs, respectively, so that they can be well adapted to the specific target image. Learning a model from a training set with image-specific adaptation in the testing stage has also been used to improve the segmentation performance. For example, an adaptive GMM has been used to address the distribution mismatch between the training and test images. For CNNs, fine-tuning is used for domainwise model adaptation to address the distribution mismatch between different training sets. However, to the best of our knowledge, this paper is the first work to propose imagespecific model adaptation for CNNs.

Method

The proposed interactive framework with Bounding box and Image-specific Fine-tuning-based Segmentation (BIFSeg) is depicted in Fig. 1. To deal with different (including previously unseen) objects in a unified framework, we propose to use a CNN that takes as input the content of a bounding box of one instance and gives a binary segmentation for that instance. In the testing stage, the user provides a bounding box, and BIFSeg extracts the region inside the bounding box and feeds it into the pre-trained CNN with a forward pass to obtain an initial segmentation. This is based on the fact that our CNNs are designed and trained to learn some common features, such as saliency, contrast and hyperintensity, across different objects, which helps to generalize to unseen objects. Then we use unsupervised (without additional user interactions) or supervised (with user-provided scribbles) image-specific fine-tuning to further refine the segmentation. This is because there is likely a mismatch between the common features learned from the training set and those in (previously unseen) test objects. Therefore, we use finetuning to leverage image-specific features and make our CNNs adaptive to a specific test image for better segmentation. Our framework is general, flexible and can handle both 2D and 3D segmentations with few assumptions of network structures. In this paper, we choose to use the state-of-the-art network structures proposed in for their compactness and efficiency. The contribution of BIFSeg is nonetheless largely different from as BIFSeg focuses on segmentation of previously unseen object classes and fine-tunes the CNN model on the fly for image-wise adaptation that can be guided by user interactions.

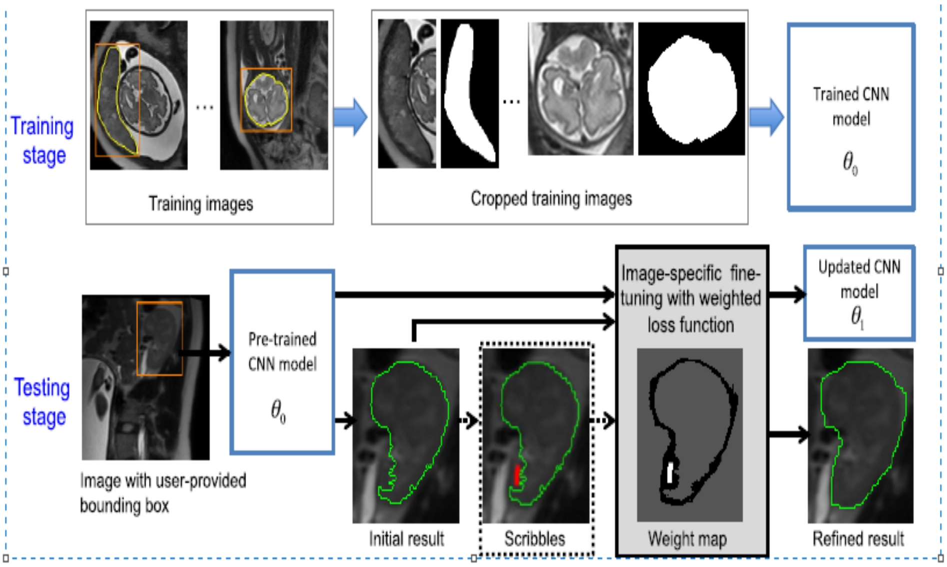


Figure 3.1: The proposed Bounding box and Image-specific Fine-tuning-based Segmentation (BIFSeg)

3.1 CNN Models

For 2D images, we adopt the P-Net for bounding box-based binary segmentation. The network is resolutionpreserving using dilated convolution. As shown in Fig. 2(a), it consists of six blocks with a receptive field of 181 × 181. The first five blocks have dilation parameters of 1, 2, 4, 8 and 16, respectively, so they capture features at different scales. Features from these five blocks are concatenated and fed into block6 that serves as a classifier. A softmax layer is used to obtain probability-like outputs. In the testing stage, we update the model based on image-specific finetuning. To ensure efficient fine-tuning and fast response to user interactions, we only fine-tune parameters of the classifier (block6). Thus, features in the concatenation layer for the test image can be stored before the fine-tuning.

For 3D images, we use a network extended from P-Net, as shown in Fig.3.1. It considers a trade-off among receptive field, inference time and memory efficiency. The network has an anisotropic[5] eceptive field of $85 \times 85 \times 9$. Compared with slice-based networks, it employs 3D contexts. Compared with large isotropic 3D receptive fields , it has less memory consumption [26]. Besides, anisotropic acquisition is often used in Magnetic Resonance (MR) imaging. We

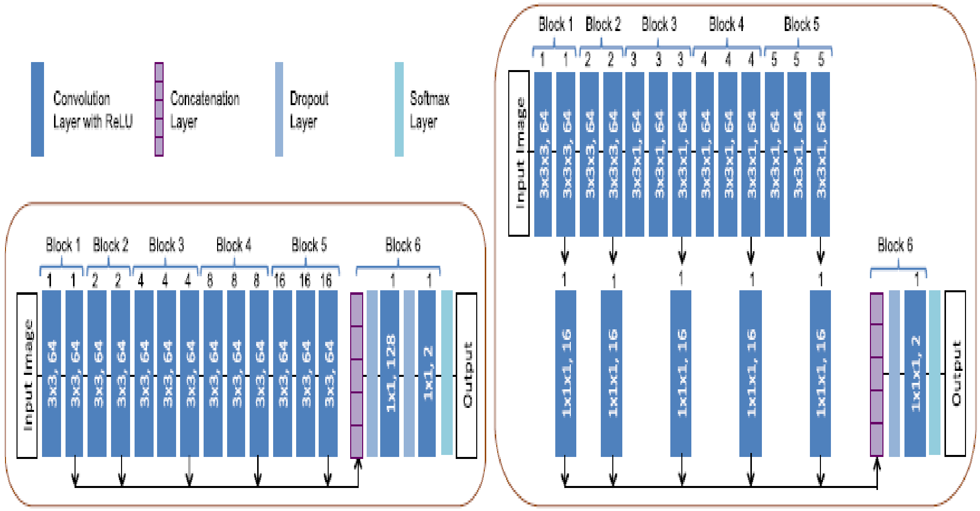


Figure 3.2: Our resolution-preserving networks with dilated convolution for 2D segmentation (a) and 3D segmentation (b)

use$3 \times 3 \times 3$ kernels in the first two blocks and $3 \times 3 \times 1$ kernels in block3 to block5. Similar to P-Net, we fine-tune the classifier (block6) with precomputed concatenated features. To save space for storing the concatenated features, we use $1{\times}1 \times 1$

$convolutions.$

3.2 Training of CNNs

Consider a K-ary segmentation training set shown in fig 3.2. T ={ (X1, Y1), (X2, Y2), . . .}where X p is one training image and Yp is the corresponding label map. The l abel set of T is {0, 1, 2,..., K 1} with 0 being the background label. LetN_k denote the number of instances of the kth object type, so the total number of instances is N = kΞN$_k$. Each image X p can have instances of multiple object classes. Suppose the label of the qth instance in X p is l pq , Yp is converted into a binary image Ypq based on whether the value of each pixel in Yp equals to l pq . The bounding box Bpq of that training instance is automatically calculated based on Ypq and expanded by a random margin in the range of 0 to 10 pixels/voxels. X p and Ypq are cropped based on Bpq. Thus, T is converted into a cropped set T^ ={(X^$_1, \hat{Y}_1), (\hat{X}_2, \hat{Y}_2), ...\}$

With size N^ and label set {0, 1} where 1 is the label of the instance foreground and 0 the background. With Tˆ, the CNN model (e.g., P-Net or PC-Net) is trained to extract the target from its bounding box, which is a binary segmentation problem irrespective of the object type. A cross entropy loss function is used for training.

3.3 Unsupervised and Supervised Image-Specific Fine-Tuning

In the testing stage, let Xˆ denote the sub-image inside a user-provided bounding box and Yˆ be the target label of Xˆ . The set of parameters of the trained CNN is . With the initial segmentation $\hat{Y0}$ obtained by the trained CNN, the user may provide (i.e., supervised) or not provide (i.e., unsupervised) a set of scribbles to guide the update of Yˆ 0. Let S f and Sb denote the scribbles for foreground and background, respectively, so the entire set of scribbles is S = S f Sb. Let si denote the user-provided label of a pixel in the scribbles, then we have si = 1 if i S f and si = 0 if i Sb. We minimize an objective function that is similar to GrabCut [20] but we use P-Net or PC-Net instead of a GMM:

arg min$_{\hat{Y},\theta}$ {E(Yˆ , $\theta) = \Xi_i\ \phi(\hat{y}_i$ X , $\theta) + \Xi_i, j\ \psi(\hat{y}_i$, ˆy $_j$ — ˆX) }

subject to : ˆy$_i$ = s$_i$ if i $\varepsilon S(1)$

is the weight of . An unconstrained optimization of an energy similar to E was used in for weakly supervised learning. In that work, the energy was based on the probability and label map of all the images in a training set, which was a different task from ours, as we focus on a single test image. We follow a typical choice of ϕ : $where[]is1if\hat{y_i}$= ˆy $_j$ and 0 otherwise. d$_ij$ is the Euclidean distance between pixel i and pixel j ,controls the effect of intensity difference. $\xi isdefinedas$:

$\phi(\hat{y}_i$— ˆX , $\theta) = -logP(\hat{y}_i$ — ˆX , $\theta) = -\hat{y}_i$logpi + (1- ˆy$_i$)log(1 - p$_i$) (2)

arg min$_{Y}$ˆ {E($\theta) = \Xi_i\ \phi(\hat{y}_i$X , $\theta) + \Xi_i, j\ \psi(\hat{y}_i$, ˆy $_i$— ˆX) }

subject to : ˆy$_i$ = s$_i$ if i $\varepsilon S(3)$

where P(ˆyi — ˆX ,) is the probability given by softmax output of the CNN, and pi = P(ˆyi = 1— ˆX ,) is the probability of pixel i belonging to the foreground. The optimization of Eq. (1) can be decomposed into steps that alternatively update the segmentation label Yˆ and network parameters . In the label update step, we fix and solve for Yˆ , and Eq. (1) becomes a Conditional Random Field (CRF) problem:

For implementation ease, the constrained optimization in Eq. (3) is converted to an unconstrained equivalent:

Since and therefore are fixed, and is submodular. In the network update step, we fix Yˆ and solve for :

arg min$_{\theta}$ $\{$E(Yˆ) =Ξ_i $\phi(\hat{y}_i$ X , $\theta)\}$

subject to : ˆy$_i$= s$_i$ if i $\varepsilon S(4)$

Thanks to the constrained optimization in Eq. (3), the label update step necessarily leads to yˆ$_i$ = s$_i$ for i S. Eq. (4) canbe treated as an unconstrained optimization:

arg min$_{\theta}$=$\{$- Ξ(ˆyi log$_p i$ + (1- ˆy$_i$)log(1 - p$_i$))$\}$ (5)

3.4 Weighted Loss Function During Network Update Step

During the network update step, the CNN is fine-tuned to fit the current segmentation Yˆ. Differently from a standard learning process that treats all the pixels equally, we propose to weight different kinds of pixels considering their confidence. First, user-provided scribbles have much higher confidence than the other pixels, and they should have a higher impact on the loss function, leading to a weighted version of Eq. (2): treated as an unconstrained optimization:

$\Xi(\hat{y}i\lceil X, \theta) = w($-i$)logP(\hat{y}_i$ — ˆX , $\theta)(6)$

where 1 is the weight associated with scribbles.   defined in Eq. (5) allows Eq. (3) to remain unchanged for the label update step. In the network update step, Eq. (8) becomes:

arg min$_{\theta}$=$\{$- Ξ(w$_i$(ˆy($_i$) log$_p i$ + (1- ˆy$_i$)log(1 - p$_i$))$\}$ (7)

Note that the energy optimization problem of Eq. (1) remains well-posed with Eq. (9), (10), and (11). Second, Yˆ may contain mis-classified pixels that can mislead the network update process. To address this problem, we propose to fine-tune the network by ignoring pixels with high uncertainty (low confidence) in the test image. We propose to use network-based uncertainty and scribblebased uncertainty. The network-based uncertainty is based on the network's softmax output. Since yˆi is highly uncertain (has low confidence) if pi is close to 0.5, we define the set of pixels with high network-based uncertainty as Up = i—t0 ¡ pi ¡ t1 where t0 and t1 are the lower and higher threshold values of foreground probability, respectively. The scribble-based uncertainty is based on the geodesic distance to scribbles. Let G(i, S f) and G(i, Sb) denote the geodesic distance [19] from pixel i to S f and Sb, respectively. Since the scribbles are drawn on mis-segmented areas for refinement, it is likely that pixels close to S have been incorrectly labeled by the initial segmentation. Let be a threshold value for the geodesic distance. We define the set of pixels with high scribble-based uncertainty as Us = U f . Therefore, a full version of the weight function is (an example is shown in Fig. 3):

The new definition of w(i) is well motivated in the network update step. However, in the label update step, introducing zero unary weights in Eq. (3) would make the label update of corresponding pixels entirely driven by the pairwise potentials. Therefore, we choose to keep Eq. (3) unchanged.

3.5 Implementation Details

We used the Caffe1 library to implement our P-Net and PC-Net.2 The training process was done via one node of the Emerald cluster3 with two 8-core E5-2623v3 Intel Haswells, a K80 NVIDIA GPU and 128GB memory. To deal with different organs and different modalities, the region inside a bounding box was normalized by the mean value and standard deviation of that region, and then used as the input of the CNNs. In the training stage, the bounding box was automatically generated based on the ground truth label with a random margin in the range of 0 to 10 pixels/voxels. We used cross entropy loss function and stochastic gradient decent with momentum 0.9, batch size 1, weight decay 5×104, maximal number of iterations 80k and initial learning 103 that was halved every 5k iterations. In the testing stage, the trained CNN models were deployed to a MacBook Pro (OS X 10.9.5) with 16GB RAM, an Intel Core i7 CPU running at 2.5GHz and an NVIDIA GeForce GT 750M GPU. A Matlab GUI and a PyQt GUI were used for user interactions on 2D and 3D images, respectively. For image-specific fine-tuning, Yˆ and were alternatively updated for four iterations. In each network update step, we used a learning rate 102 and iteration number 20. We used a grid search with the training data to get proper values of , , t0, t1, and , and fixed them as global parameters during testing. Their numerical values are listed in the specific experimental sections III-B and III-C.

3.6 Performance Evaluation

We validated the proposed framework with two applications: 2D segmentation of multiple organs from fetal MRI and 3D segmentation of brain tumors from contrast enhanced T1-weighted (T1c) and Fluid-attenuated Inversion Recovery (FLAIR) images. For both applications, we additionally investigated the segmentation performance on previously unseen objects that were not present in the training set.

3.6.1 Comparison Methods and Evaluation Metrics

To investigate the performance of different networks with the same bounding box, we compared P-Net with FCN and U-Net for 2D images, and compared PC-Net with DeepMedic and HighRes3DNet for3D images.4 The original DeepMedic works on multiple modalities, and we adapted it to work on a single modality. All these methods were evaluated on the laptop during the testing except for HighRes3DNet that was run on the cluster due to the laptop's limited GPU memory. To validate the proposed unsupervised/supervised image-specific fine-tuning, we compared BIFSeg with 1) the initial output of P-Net/ PC-Net, 2) post-processing the initial output with a CRF (using user interactions as hard constraints if they were provided), and 3) image-specific fine-tuning based on Eq. (1) with w(i) = 1 for all the pixels, which is referred to as BIFSeg(-w).

BIFSeg was also compared with other interactive methods: GrabCut , Slic-Seg and Random Walks for 2D segmentation, and GeoS , GrowCut and 3D GrabCut [29] for 3D segmentation. The 2D/3D GrabCut used the same bounding box as used by BIFSeg, and they used 3 and 5 components for the foreground and background GMMs, respectively. Slic-Seg, Random Walks, GeoS and GrowCut required scribbles without a bounding box for segmentation. The segmentation results by an Obstetrician and a Radiologist were used for evaluation. For each method, each user provided scribbles to update the result multiple times until the user accepted it as the final segmentation. The Dice score between a segmentation and the ground truth was used for quantitative evaluations: Dice = 2—Ra Rb—/(—Ra—+—Rb—) where Ra and Rb denote the region segmented by an algorithm and the ground truth, respectively. We used a paired Student's t-test to determine whether the performance difference between two segmentation methods was significant [30]. The p-value, i.e., the probability of achieving a more extreme value than the observed segmentation performance difference, when the null hypothesis is true, was calculated for significance assessment.

3.6.2 2D Segmentation of Multiple Organs From Fetal MRI

1) Data: Single-shot Fast Spin Echo (SSFSE) was used to acquire stacks of T2-weighted MR images from 18 patients with pixel size 0.74 to 1.58 mm and inter-slice spacing 3 to 4 mm. Due to the large inter-slice spacing and inter-slice motion, interactive 2D segmentation is more suitable than direct 3D segmentation. We performed data splitting at patient level and used images from 10, 2, 6 patients for training, validation and testing, respectively. The training set consisted of 333 and 213 2D instances of the placenta and fetal brain, respectively. The validation set contained 70, 25, 36 and 41 2D instances of the placenta, fetal brain, fetal lungs and maternal kidneys, respectively. The testing set consisted of 165, 80, 114 and 124 2D instances of the placenta, fetal brain, fetal lungs and maternal kidneys, respectively. Here the fetal brain and the placenta were previously seen objects, and the fetal lungs and maternal kidneys were previously unseen objects. Manual segmentations by a Radiologist were used as the ground truth. The P-Net was used for this segmentation task. The bounding boxes of organs in the training set had an original side length of 98±59 pixels. To deal with organs at different scales, we resized the input of P-Net so that the minimal value of width and height was 96 pixels. In the testing stage, the output of BIFSeg for one object was resized to fit its bounding box in the original image. Parameter setting was = 3.0, = 0.1, t0 = 0.2, t1 = 0.7, = 0.2, = 5.0 based on a grid search with the training data (i.e., fetal lungs and maternal kidneys were not used for parameter learning).

3.6.3 Initial Segmentation Based on P-Net

Fig. 3.3 presents the evolution of the loss on the training and validation data with FCN, U-Net and P-Net during the training stage. It shows that FCN and U-Net tend to over-

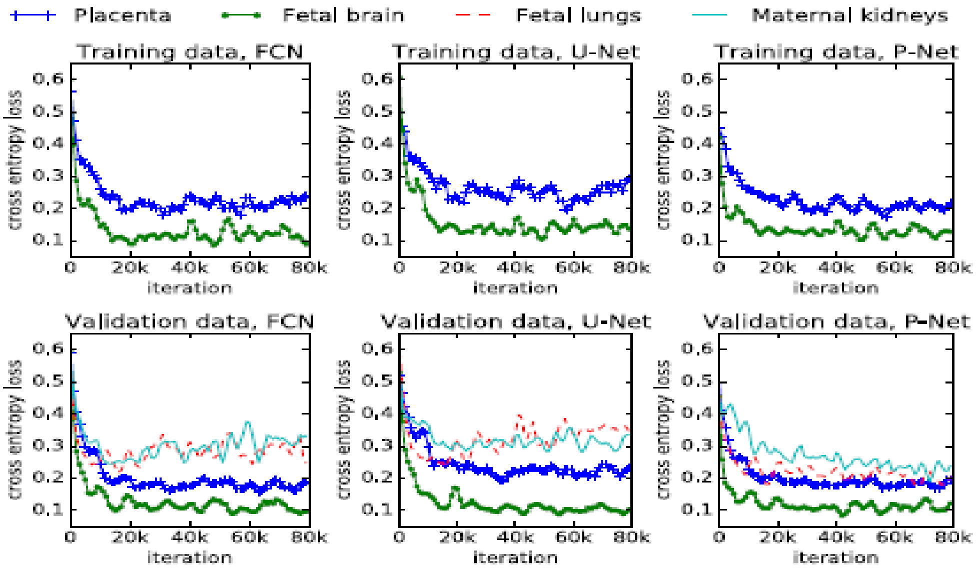


Figure 3.3: Evolution of cross entropy loss on training and validation data during the training stage of different networks for 2D fetal MRI segmentation.

fit the placenta and fetal brain in the training set, while P-Net generalizes better to previously unseen fetal lungs and maternal kidneys in comparison. Fig. 3.4 shows the initial segmentation of different organs from fetal MRI with user-provided bounding boxes. It can be observed that GrabCut achieves a poor segmentation except for the fetal brain where there is a good contrast between the target and the background. For the placenta and fetal brain, FCN, U-Net and P-Net achieve visually similar results that are close to the ground truth. However, for fetal lungs and maternal kidneys that are previously unseen in the training set, FCN and U-Net lead to a large region of under-segmentation. In contrast, P-Net performs noticeably better than FCN and U-Net when dealing with these two unseen objects. A quantitative evaluation of these methods is listed in Table I. It shows that P-Net achieves the best accuracy for unseen fetal lungs and maternal kidneys with average machine time 0.16s.

3.6.4 Unsupervised Image-Specific Fine-Tuning

For unsupervised refinement, the initial segmentation obtained by P-Net was refined by CRF, BIFSeg(-w) and BIFSeg without additional scribbles, respectively. The results are shown in Fig. 3.4. The second to fourth rows show the foreground probability obtained by

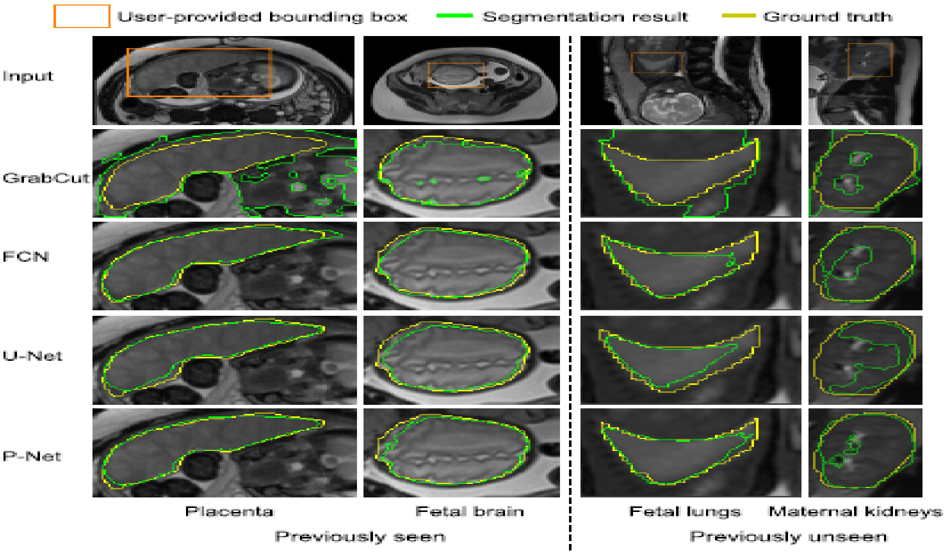


Figure 3.4: Visual comparison of initial segmentation of multiple organs from fetal MRI with a bounding box.

P-Net before and after the fine-tuning. In the second row, the initial output of P-Net has a probability around 0.5 for many pixels, which indicates a high uncertainty. After image-specific fine-tuning, most pixels in the outputs of BIFSeg(-w) and BIFSeg have a probability close to 0.0 or 1.0. The remaining rows show the outputs of P-Net and the three refinement methods, respectively. The visual comparison shows that BIFSeg performs better than P-Net + CRF and BIFSeg(-w). Quantitative measurements are presented in Table II. It shows that BIFSeg achieves a larger improvement of accuracy from the initial segmentation when compared with the use of CRF or BIFSeg(-w). In this 2D case, BIFSeg takes 0.72s in average for unsupervised image-specific fine-tuning

3.6.5 Supervised Image-Specific Fine-Tuning

Fig. 3.5 shows examples of supervised refinement with additional scribbles. The same initial segmentation and scribbles are used for P-Net + CRF, BIFSeg(-w) and BIFSeg. All these methods improve the segmentation. However, some large mis-segmentations can still be observed for P-Net + CRF and BIFSeg(-w). In contrast, BIFSeg achieves better results with the same set of scribbles. For a quantitative comparison, we measured the segmentation

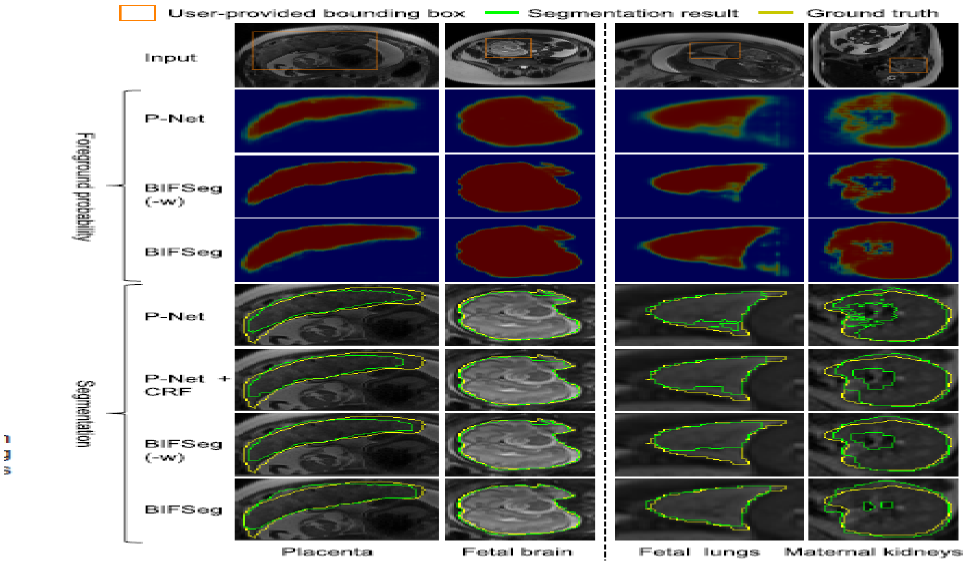


Figure 3.5: Visual comparison of P-Net and three unsupervised refinement methods for fetal MRI segmentation.

accuracy after a single round of refinement using the same set of scribbles. The result is shown in Table III. BIFSeg achieves significantly better

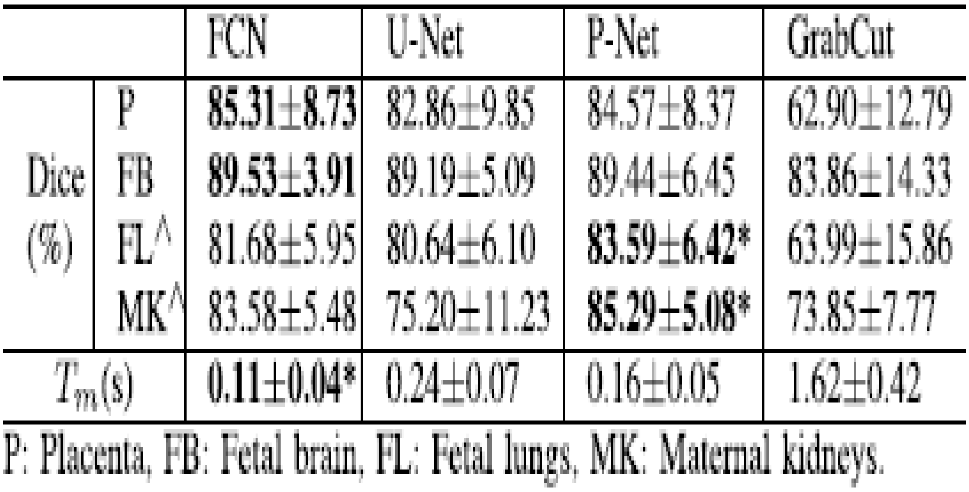
accuracy (p-value ¡ 0.05) for the placenta, and previously unseen fetal lungs and maternal kidneys compared with P-Net + CRF and BIFSeg(-w). Fig. 3.6 shows a visual comparison of unsupervised and supervised fine-tuning of BIFSeg for the same maternal kidney. Table II and Table III show that supervised fine-tuning achieves 3-5 percentage points higher Dice than unsupervised fine-tuning

3.6.6 Comparison With Other Interactive Methods

The two users (an Obstetrician and a Radiologist) used Slic-Seg , GrabCut , Random Walks and BIFSeg for the fetal MRI segmentation tasks respectively. For each image, the segmentationwas refined interactively until it was accepted by Fig 3.7 Quantitative comparison on initial fetal MRI segmentation from a bounding box .Tm is the machine time $\wedge$ denotes previously unseen objects.

section3D Segmentation of Brain Tumors From T1c and FLAIR

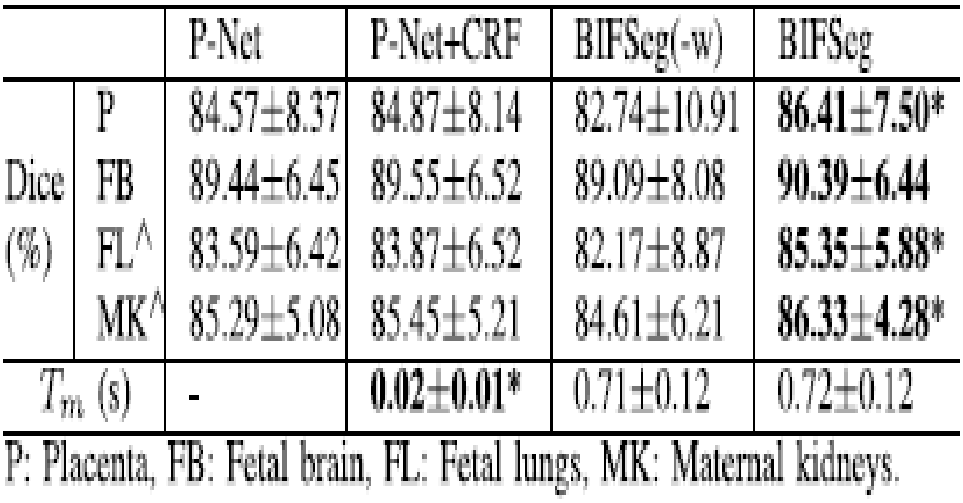
We used the 2015 Brain Tumor Segmentation Challenge training set. The ground truth



		FCN	U-Net	P-Net	GrabCut
Dice	P	**85.31±8.73**	82.86±9.85	84.57±8.37	62.90±12.79
	FB	**89.53±3.91**	89.19±5.09	89.44±6.45	83.86±14.33
(%)	FL^	81.68±5.95	80.64±6.10	**83.59±6.42***	63.99±15.86
	MK^	83.58±5.48	75.20±11.23	**85.29±5.08***	73.85±7.77
T_m(s)		**0.11±0.04***	0.24±0.07	0.16±0.05	1.62±0.42

P: Placenta, FB: Fetal brain, FL: Fetal lungs, MK: Maternal kidneys.

Table 3.1: Visual comparison of P-Net and three unsupervised refinement methods for fetal MRI segmentation.



		P-Net	P-Net+CRF	BIFSeg(-w)	BIFSeg
	P	84.57±8.37	84.87±8.14	82.74±10.91	**86.41±7.50***
Dice	FB	89.44±6.45	89.55±6.52	89.09±8.08	**90.39±6.44**
(%)	FL^	83.59±6.42	83.87±6.52	82.17±8.87	**85.35±5.88***
	MK^	85.29±5.08	85.45±5.21	84.61±6.21	**86.33±4.28***
T_m (s)		-	**0.02±0.01***	0.71±0.12	0.72±0.12

P: Placenta, FB: Fetal brain, FL: Fetal lungs, MK: Maternal kidneys.

Table 3.2: Visual comparison of P-Net and three unsupervised refinement methods for fetal MRI segmentation.

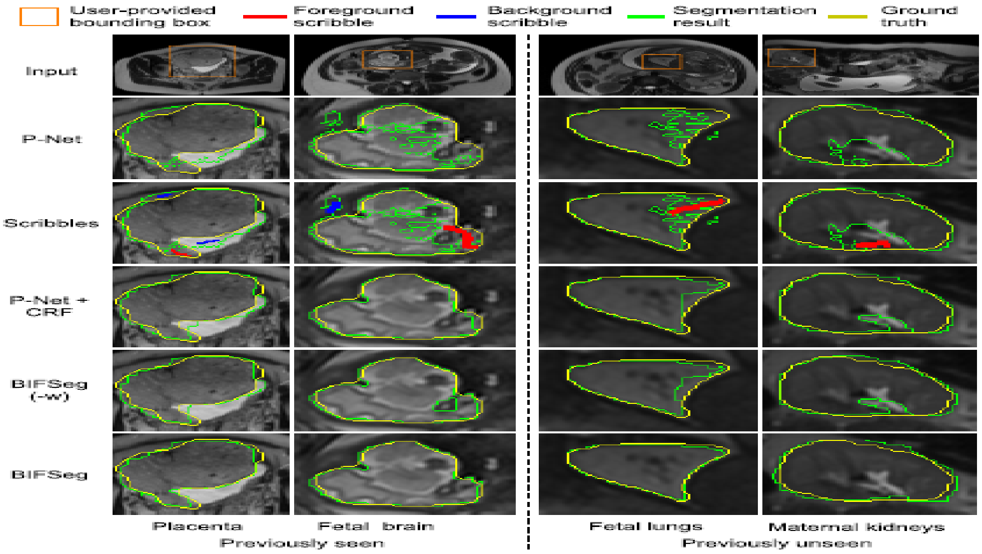


Figure 3.6: Visual comparison of P-Net and three supervised refinement methods for fetal MRI segmentation.

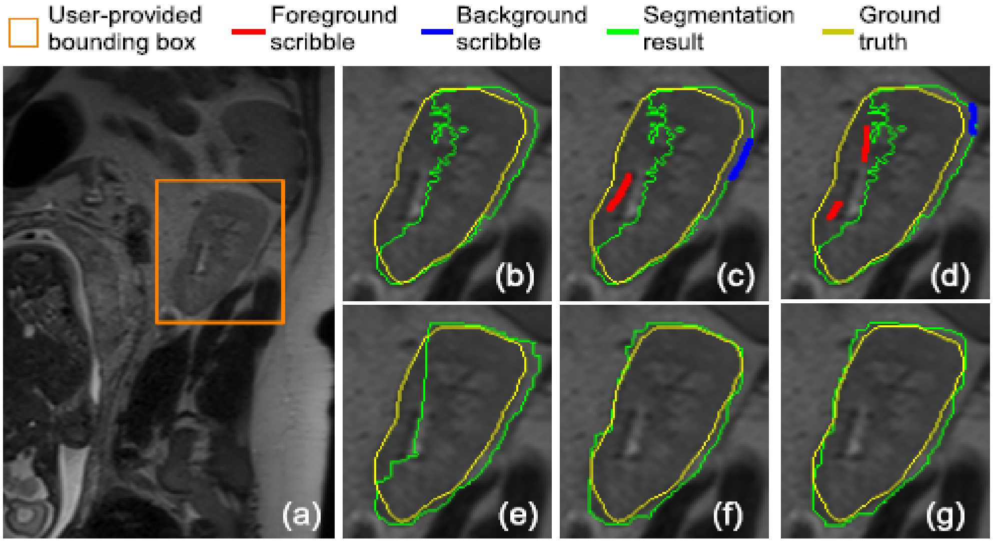
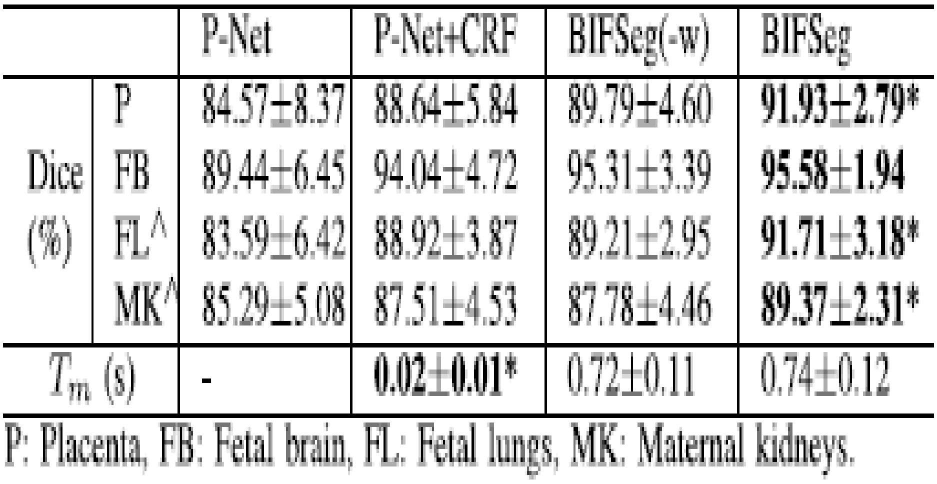


Figure 3.7: Unsupervised and supervised fine-tuning results of BIFSeg for the same instance of previously unseen maternal kidneys.



		P-Net	P-Net+CRF	BIFSeg(-w)	BIFSeg
Dice (%)	P	84.57±8.37	88.64±5.84	89.79±4.60	**91.93±2.79***
	FB	89.44±6.45	94.04±4.72	95.31±3.39	**95.58±1.94**
	FL^	83.59±6.42	88.92±3.87	89.21±2.95	**91.71±3.18***
	MK^	85.29±5.08	87.51±4.53	87.78±4.46	**89.37±2.31***
T_m (s)		-	**0.02±0.01***	0.72±0.11	0.74±0.12

P: Placenta, FB: Fetal brain, FL: Fetal lungs, MK: Maternal kidneys.

Table 3.3: User time and Dice score of different interactive methods for fetal MRI segmentation

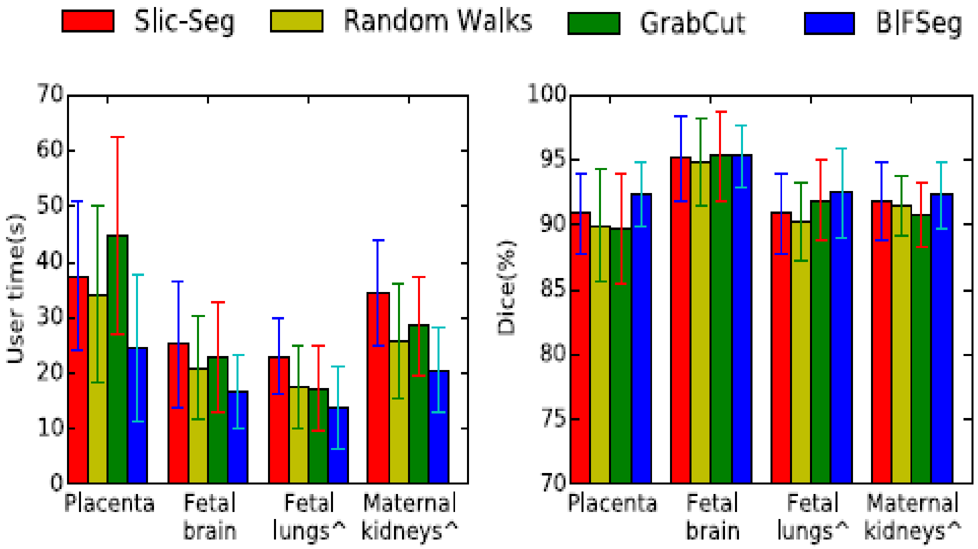


Figure 3.8: Our resolution-preserving networks with dilated convolution for 2D segmentation (a) and 3D segmentation (b)

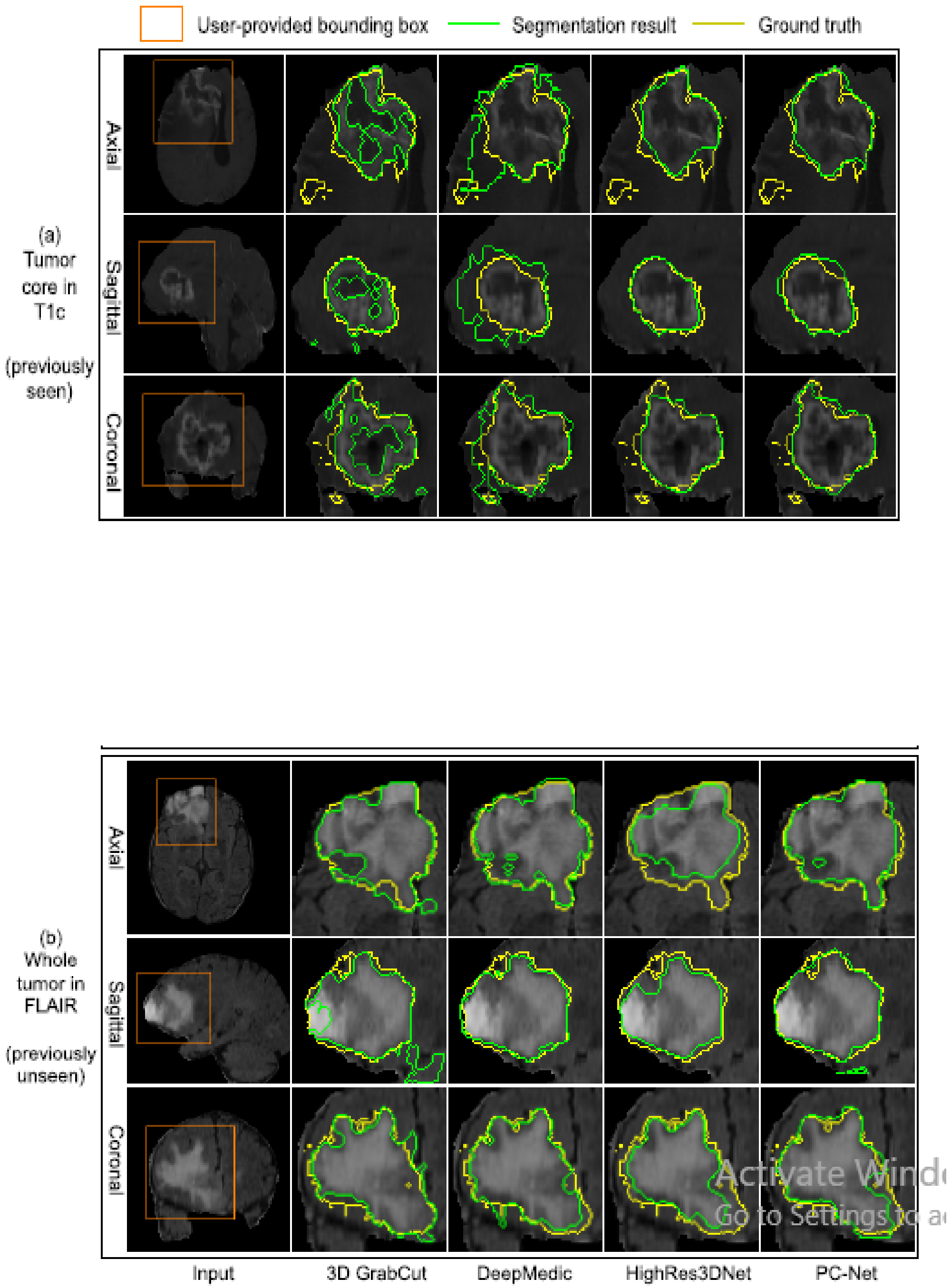


Figure 3.9: Our resolution-preserving networks with dilated convolution for 2D segmentation (a) and 3D segmentation (b)

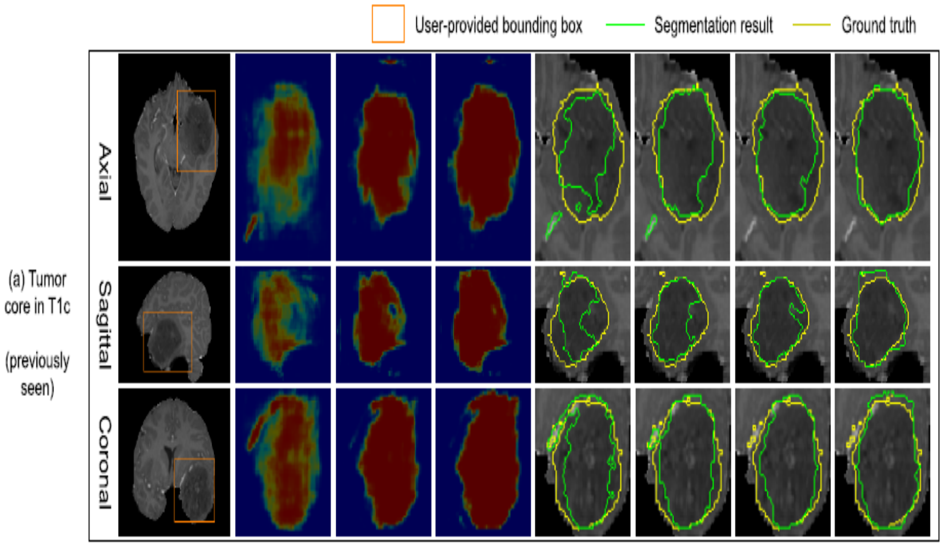


Figure 3.10: Visual comparison of PC-Net and unsupervised refinement methods without additional scribbles for 3D brain tumor segmentation.

were manually delineated by experts. This dataset included

274 scans from 198 patients. Each scan used multiple MR sequences with different contrasts. T1c highlights the tumor without peritumoral edema, designated "tumor core" as per . FLAIR highlights the tumor with peritumoral edema, designated "whole tumor" as per . We investigate interactive segmentation of the tumor core from T1c images and the whole tumor from FLAIR images, which is different from previous works on automatic multi-label and multi-modal segmentation. We randomly selected T1c and FLAIR images of 19, 25 patients with a single scan for validation and testing, respectively, and used T1c images of the remaining patients for training. Here the tumor core in T1c images was previously seen while the whole tumor in FLAIR images was previously unseen for the CNNs. All these images had been skull-stripped and resampled to isotropic 1mm3 resolution. The maximal side length of bounding boxes of the tumor core and the whole tumor ranged from 40 to 100 voxels, we resized the cropped image region inside a bounding box so that its maximal side length was 80 voxels. Parameter setting was = 10.0, = 0.1, t0 = 0.2, t1 = 0.6, = 0.2, = 5.0 based on a grid search with the training data (i.e., whole tumor images were not used for parameter learning).

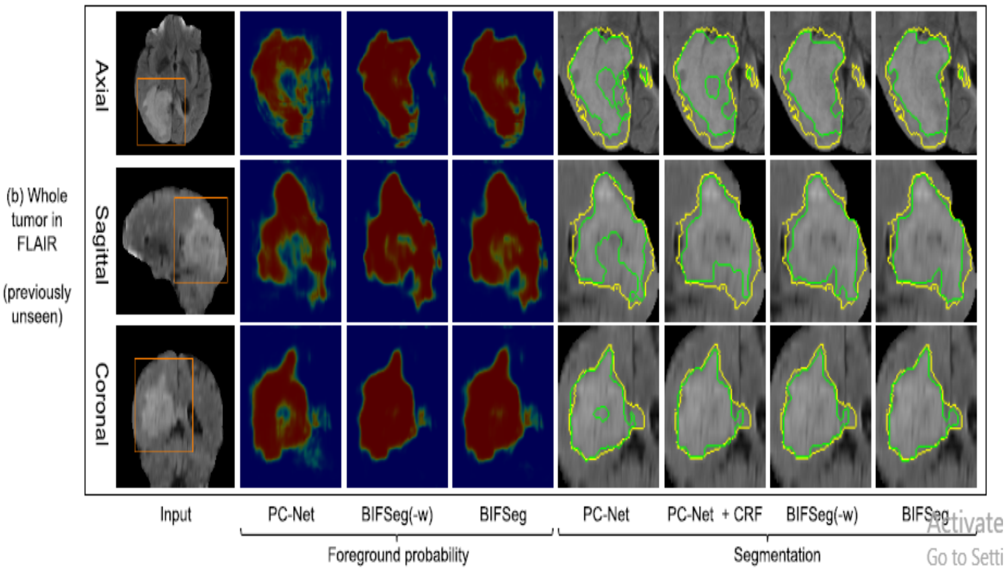


Figure 3.11: Visual comparison of PC-Net and unsupervised refinement methods without additional scribbles for 3D brain tumor segmentation.

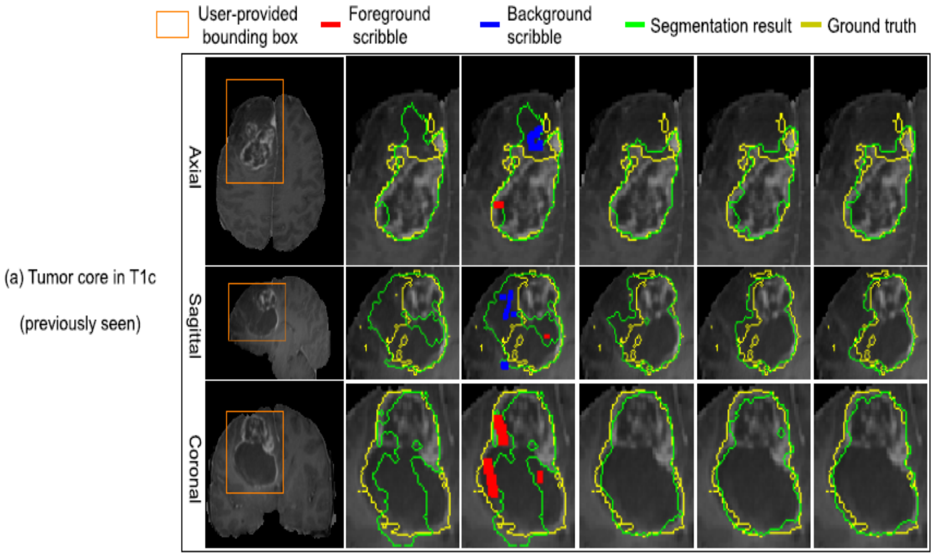


Figure 3.12: Visual comparison of PC-Net and unsupervised refinement methods without additional scribbles for 3D brain tumor segmentation.

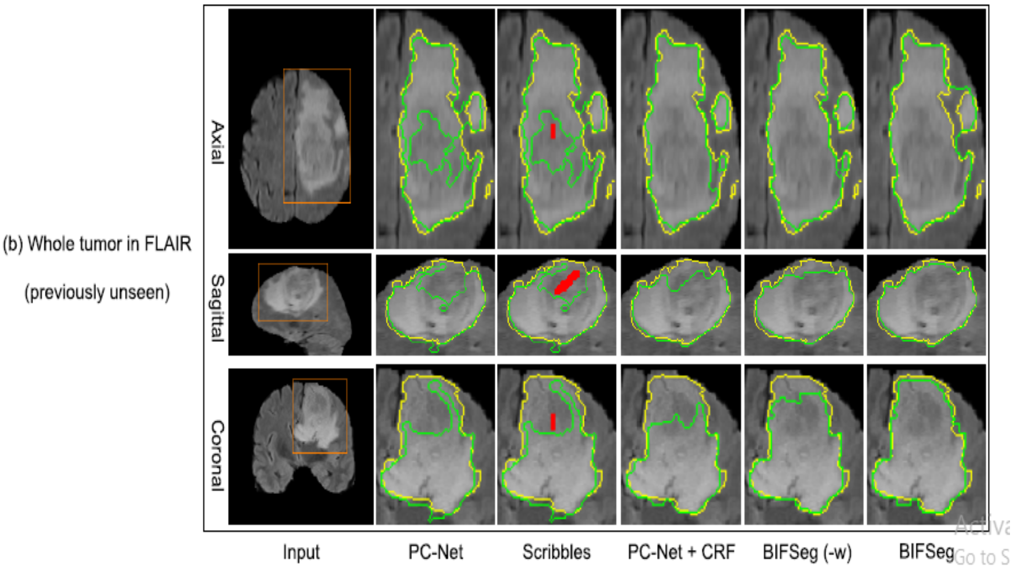


Figure 3.13: Visual comparison of PC-Net and unsupervised refinement methods without additional scribbles for 3D brain tumor segmentation.

3.6.7 Initial Segmentation Based on PC-Net

: Fig. 3.12 shows an initial result of tumor core segmentation from T1c with a user-provided bounding box. Since the central region of the tumor has a low intensity that is similar to the background, 3D GrabCut obtains large under-segmentations. DeepMedic leads to some over-segmentations. HighRes3DNet and PC-Net obtain similar results, but PC-Net is less complex and has lower memory consumption. Fig. 3.13 shows an initial segmentation result of previously unseen whole tumor from FLAIR. 3D GrabCut fails to get high accuracy due to intensity inconsistency in the tumor region, and the CNNs outperform 3D GrabCut, with DeepMedic and PC-Net performing better than HighRes3DNet. A quantitative comparison is presented in Table IV. It shows that the performance of DeepMedic is low for T1c but high for FLAIR, and that of HighRes3DNet is the opposite. This is because DeepMedic has a small receptive field and tends to rely on local features. It is difficult to use local features to deal with T1c due to its complex appearance but easier to deal with FLAIR since the appearance is less complex. HighRes3DNet has a more complex model and tends to over-fit the tumor core. In contrast, PC-Net achieves a more stable performance on the tumor core and the previously unseen whole tumor. The average machine time for 3D GrabCut, DeepMedic, and PC-Net is

3.87s, 65.31s and 3.83s, respectively (on the laptop), and that for HighRes3DNet is 1.10s (on the cluster).

3.6.8 Unsupervised Image Specific Fine Tuning

Figure 3.13 shows unsupervised fine-tuning for brain tumor segmentation without additional user interactions. In Fig. 3.14, the tumor core is under-segmented in the initial output of PC-Net. CRF improves the segmentation to some degree, but large areas of under-segmentation still exist. The segmentation result of BIFSeg(-w) is similar to that of CRF. In contrast, BIFSeg performs better than CRF and BIFSeg(-w). A similar situation is observed in Fig. 3.14(b) for segmentation of previously unseen whole tumor.

3.6.9 Supervised Image-Specific Fine-Tuning

Fig 3.15 shows refined results of brain tumor segmentation with additional scribbles provided by the user. The same initial segmentation based on PC-Net and the same scribbles are used by CRF, BIFSeg(-w) and BIFSeg. It can be observed that CRF and BIFSeg(-w) correct the initial segmentation moderately. In contrast, BIFSeg achieves better refined results for both the tumor core in T1c and the whole tumor in FLAIR. For a quantitative comparison, we measured the segmentation accuracy after a single round of refinement using the same set of scribbles based on the same initial segmentation. The result is presented in Table VI, showing BIFSeg significantly outperforms CRF and BIFSeg(-w) in terms of Dice. Table V and Table VI show that supervised fine-tuning achieves 1.3-1.8 percentage points higher Dice than unsupervised fine-tuning for brain tumor segmentation.

3.6.10 Comparison With Other Interactive Methods

The two users (an Obstetrician and a Radiologist) used GeoS, GrowCut [6], 3D GrabCut and BIFSeg for the brain tumor segmentation tasks respectively. The user time and final accuracy of these methods are presented in Fig. 3.16. It shows that these interactive methods achieve similar final Dice scores for each task. However, BIFSeg takes significantly less user time, which is 82.3s and 68.0s in average for the tumor core and the whole tumor, respectively.

Conclusion

Applying pre-trained models to previously unseen objects is a zero-shot learning problem [7]. While previous works studied zero-shot learning for image classification [8], this paper focused on the context of medical image segmentation. For 2D images, our P-Net was trained with the placenta and fetal brain only, but it performed well on previously unseen fetal lungs and maternal kidneys. There are two main reasons for this. First, these four organs were imaged with the same protocol. They have similar signal to noise ratio and share some common features, such as saliency, contrast and hyperintensity. Second, compared with FCN and U-Net, P-Net has far fewer parameters without reduction of the receptive field. Therefore, it can generalize better to previously unseen objects. Similarly, the tumor core and whole tumor have some common features, e.g., lower or higher intensity than the remaining brain regions. PC-Net is more compact than HighRes3DNet and less likely to achieve over-fitting, leading to better ability to deal with the unseen whole tumor. Our BIFSeg framework is theoretically applicable to different CNN models. However, this research focuses on interactive segmentation, where short inference time and memory efficiency of the network are key requirements to enable responsive user interfaces and to work on machines with limited GPU resources. This is especially critical for 3D image segmentation. DeepMedic takes over 60 seconds for inference, while HighRes3DNet has too large a memory consumption to work on a laptop. They are thus less suitable for interactive segmentation compared with PC-Net. We have designed PC-Net with the explicit requirement of interactive runtime on a laptop. In this paper we only used at most two objects in the training set. To further increase BIFSeg's ability to generalize, it is of interest to use a larger training set with more patients.

REFERENCES

[1] F. Zhao and X. Xie, “An overview of interactive medical image segmentation,” Ann. BMVA, vol. 2013, no. 7, pp. 1–22, 2013.

[2] K. Kamnitsas et al., “Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation,” Med. Image Anal., vol. 36, pp. 61–78, Feb. 2017.

[3] F. Milletari, N. Navab, and S.-A. Ahmadi, “V-net: Fully convolutional neural networks for volumetric medical image segmentation,” in Proc. IC3DV, Oct. 2016, pp. 565–571.

[4] D. Lin, J. Dai, J. Jia, K. He, and J. Sun, “ScribbleSup: Scribblesupervised convolutional networks for semantic segmentation,” in Proc. CVPR, Jun. 2016, pp. 3159–3167.

[5] V. Vezhnevets and V. Konouchine, “GrowCut: Interactive multi-label ND image segmentation by cellular automata,” in Proc. Graphicon, 2005, pp. 150–156.

[6] Z. Zhang and V. Saligrama, “Zero-shot learning via semantic similarity embedding,” in Proc. ICCV, 2015, pp. 4166–4174.

[7] Y. Fu, T. M. Hospedales, T. Xiang, and S. Gong, “Transductive multiview zero-shot learning,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 37, no. 11, pp. 2332–2345, Nov. 2015.

[8] K. Keraudren et al., “Automated fetal brain segmentation from 2D MRI slices for motion correction,” NeuroImage, vol. 101, pp. 633–643, Jul. 2014.