An Efficient Interactive Multi-label Segmentation Tool for 2D and 3D Medical Images using Fully Connected Conditional Random Field

Ruizhe Li[a], Xin Chen[a]

[a]Intelligent Modelling and Analysis Group, School of Computer Science, University of Nottingham, UK.

ARTICLE INFO

Keywords:
2D&3D Medical Image Segmentation
Conditional Random Filed

ABSTRACT

Objective: Image segmentation is a crucial and fundamental step in many medical image analysis tasks, such as tumor measurement, surgery planning, disease diagnosis, etc. To ensure the quality of image segmentation, most of the current solutions require labor-intensive manual processes by tracing the boundaries of the objects. The workload increases tremendously for the case of three dimensional (3D) image with multiple objects to be segmented.
Method: In this paper, we introduce our developed interactive image segmentation tool that provides efficient segmentation of multiple labels for both 2D and 3D medical images. The core segmentation method is based on a fast implementation of the fully connected conditional random field. The software also enables automatic recommendation of the next slice to be annotated in 3D, leading to a higher efficiency.
Results: We have evaluated the tool on many 2D and 3D medical image modalities (e.g. CT, MRI, ultrasound, X-ray, etc.) and different objects of interest (abdominal organs, tumor, bones, etc.), in terms of segmentation accuracy, repeatability and computational time.
Conclusion: In contrast to other interactive image segmentation tools, our software produces high quality image segmentation results without the requirement of parameter tuning for each application. Both the software and source code are freely available for research purpose[1].

1. Introduction

Image segmentation has been an active research topic for a few decades in the areas of computer vision, medical image analysis and etc. In our work, we focus on the application of image segmentation in medical image analysis tasks, such as tumor quantification, surgery planning and disease diagnosis. There are a large number of research papers published in the topic of medical image segmentation, which can be categorized based on different criteria. In this paper, from the user point of view, we classify different methods into manual, semi-automatic and fully automatic.

Many commercial and open-source software offer manual delineations for medical images, including line tracing, polynomial curve fitting, area painting, etc. ITK-Snap [1] is an open-source and widely used tool that is mainly dedicated to medical image segmentation. It offers polygon and paintbrush tools for flexible editing of both 2D and 3D images. Another widely acknowledged open-source tool is 3D Slicer [2]. It provides manual tools such as area painting, level tracing and scissors, which are normally used as post-processing to refine segmentation results using threshold or region growing. These manual tools offer good quality control but require tremendous time and effort from the user.

There are many classical fully automatic methods [3], such as Ostu's threshold, K-means clustering, etc. These automatic methods are normally application dependent, which can not work robustly unless the object of interest has a homogeneous image intensity and well distinguishable from the other image regions. Many of these methods nowadays are used as a pre-processing step of other more sophisticated methods, such as region of interest extraction for removing redundant information and initial labeling for weakly supervised machine learning methods. Recently, deep learning methods have achieved the state-of-the-art performance in many image segmentation tasks [4], which are fully automatic in inference time. However, they often require a large number of annotated images for supervised model training, which require either manual or semi-automatic image segmentation to generate the training dataset.

Semi-automatic methods take the advantage of automatic segmentation and allow users to intervene with the segmentation process. One type of user interaction is initialization, such as drawing seeds or bounding boxes inside or around the target object. Then the seeds or initial contour evolve to the desired object's boundary by region growing [5] or minimizing an energy function (e.g. active contour [6], level sets [7], etc.). These methods do not offer post-segmentation user interactions to further refine the results and the parameter settings are highly application dependent. Another type of user interaction is to iteratively improve the segmentation results by adding scribbles to different classes (e.g grow cut [8], graph cut [9], etc.). At each iteration, the method propagates these labels to the whole image by optimizing an energy function. This is more or less guaranteed to achieve a satisfactory result with reduced workload compared to a manual process, which is desirable for medical image segmentation.

Our proposed solution is based on user interactions using scribbles, and the image segmentation task is converted into a graph optimization problem using conditional random field. We introduce related work in the literature and high-

[1]Software download: http://www.cs.nott.ac.uk/~pszxc/ (password: iseg2020).

ruizhe.li@nottingham.ac.uk (R. Li); xin.chen@nottingham.ac.uk (X. Chen)

light our contributions in section 1.1 and section 1.2 respectively. In section 2, detailed methodology of our method is described. The parameter setting and graphical user interface are introduced in section 3. Evaluation results are presented in section 4, followed by discussion and conclusions in section 5.

1.1. Related Work

Our method is based on an interactive strategy that dynamically react to the user's annotations. Hence we focus on discussing related work in this category. There are many types of user annotations, such as point-based, contour-based, scribble-based and bounding boxbased methods. We adopt the scribble-based interaction approach due to its high flexibility and user-friendly nature.

In our implementation, the image to be segmented is considered as a graph. User's annotations serve as a prior knowledge to determine the likelihood of individual pixel belonging to each of the labeled classes. Together with this prior information, the pixel-wise similarity and label consistency are normally modeled by Markov random filed or conditional random field (CRF). The image segmentation task is then converted into an energy optimization problem over a graph structure. Although exact inference of such a structure is intractable, a lot of efforts have been made to develop approximation algorithms, including iterated conditional modes [10], belief propagation [11], max-flow/min-cut [12] and filter-based inference [13], in which filterbased mean field inference and graph cut are the two most popular solutions.

Boykov et al. [9] proposed the two mostly used graph cut algorithms: $\alpha\beta$-swap and α-expansion. In $\alpha\beta$-swap, for a pair of labels α and β, it exchanges the labels between an arbitrary set of pixels labeled α and another arbitrary set labeled β. The algorithm generates a labeling such that there is no swap move that decreases energy in the predefined graph. The $\alpha\beta$-swap method works well for a binary graph (two-labels) but difficult to be extended to multiclass segmentation. Alternatively, α-expansion is suitable for a multiclass problem. It starts with any labeling and runs through all labels iteratively. For each label α, it computes an optimal α-expansion move and accepts the move if the energy decreases. The algorithm is terminated when there is no expansion move that decreases the energy. Graph cut method has been applied to interactive image segmentation by Rother et al. [14], called "grab cut". In grab cut, users only need to draw a bounding box around the object of interest, the foreground object is then segmented using graph cut. The segmentation result can be further refined using additional scribbles. Grab cut works superbly with minimal user input, but it is limited to binary class segmentation and the computational speed is slow in 3D. Kohli et al. [15] extended the class of energy functions for which the optimal α-expansion and $\alpha\beta$-swap moves can be computed in polynomial time. However, the inference speed and memory usage is still inefficient comparing to the mean field inference method, especially when there are multiple labels in 3D images.

Many methods of mean field approximations in computer vision have been proposed, such as object class segmentation [16]. The mean field algorithm approximates the exact distribution P using a distribution Q calculated as a product of independent marginal by minimizing the KL divergence $D(Q|P)$. Although the approximation of P as a fully factored distribution is likely to lose some information in the distribution, this approximation is computationally efficient. Krähenbühl et. al. [13] developed a filter-based method for performing fast fully connected CRF optimization, which is the core algorithm used in our work.

More recently, many image segmentation methods have been proposed based on deep convolutional neural networks (DCNN). Majority of the methods are fully automatic but require a large number of annotated labels for fully supervised learning. There are a few work that allow user interactions. DeepCut [17] utilises patch based DCNN and CRF to segment objects from bounding boxes. It only works for a binary class problem. Similarly, ScribbleSup [18] trains DCNN to learn and perform multi-object segmentation from user annotated scribbles. Both methods do not allow dynamic user interactions to iteratively refine the result. Wang et al. [19] proposed an interactive image segmentation method based on DCNNs. In their method, an initial segmentation result is obtained using fully supervised DCNN model. Another DCNN model is applied to take user interactions for refining the initial result in a CRF structure. The implementation is currently limited to binary labels. The main drawback of these DCNN-based methods is the requirement of a training process using either weakly or fully labeled data.

1.2. Contributions

In summary, the key issues with the current interactive segmentation solutions for medical images are three folds. (1) Many image segmentation tools in computer vision work well in 2D images, but not many of them are applicable to 3D medical images, where the key barrier is computational time and effectiveness of user interactions in 3D. (2) Many solutions only focus on binary image segmentation, while multiple organs are often required to be segmented in medical images. (3) Many generic interactive image segmentation methods often work well in natural images with rich regional textures, which may not be directly applicable to medical images where multiple labels need to be assigned to image regions that have similar intensities. For example, carpel bone segmentation in the wrist [20]. Moreover, different parameter settings are normally required for different images. We aim to produce a generic medical image segmentation tool that works for both 2D and 3D images without any prior information or training process, and allows efficient user interactions.

To achieve the aim and address the aforementioned issues, the key contributions of our work are summarized as follows. (1) A fast CRF solver based on Gaussian approximation is adopted to achieve fast 2D and 3D image segmentation for multiple labels (up to 10 labels in the current implementation and can be easily extended.). (2) The software

is featured with an automatic slice recommendation function to suggest the best slice to annotate, resulting in greatly improved image segmentation efficiency in 3D. (3) We have optimally tuned the parameters in the algorithm, so that an "one size for all" setting is achieved, meaning no parameter adjustment is required for different medical image segmentation tasks. The developed tool has been evaluated on a variety of 2D and 3D medical image modalities and applications, in terms of segmentation accuracy, repeatability and computational time.

2. Methodology

The goal of image segmentation is to label every pixel in the image with one of several predetermined object categories. In this paper, it is formulated as maximum a posteriori (MAP) inference in a CRF, which is defined over pixels in an image. The object classes (categories) are defined interactively by the user using scribbles. The CRF is constructed by combining a smoothness term that maximizes the label agreement between similar pixels and the likelihood of each pixel belonging to each of the user defined classes. The CRF is dynamically changed when more scribbles are added, leading to an iteratively refined segmentation result. Our aim is to minimize the user interactions while achieving high quality image segmentation.

2.1. Fully Connected Conditional Random Field

In this section, we introduce how the image segmentation task is formulated by CRF optimization. For an image, the CRF model can be constructed as: $I = \{I_1, ..., I_N\}$ that represent the intensities of N pixels, and the random field $X = \{X_1, ..., X_N\}$ that indicate possible pixel-wise labeling. The domain of X is a set of labels $L = \{0, ..., K-1\}$ in K classes. For a binary class segmentation, $K = 2$ and $L = \{0, 1\}$. A configuration x is one of the possible assignments of all N pixels and the ground truth labels (dented as y) is one of them.

Based on Gibbs distribution as described below,

$$P(X|I) = \frac{1}{Z(I)} exp(-\sum_{c \in C_G} \phi_c(X_c|I)) \quad (1)$$

A graph $G = (V, E)$ on X is defined via potential ϕ_c, where c is a clique that belongs to a set of cliques C_G in the graph G. V and E are the vertices and edges of the graph respectively. $Z(I)$ is the partition function that normalizes the distribution. The Gibbs energy of a configuration $x \in L^N$ is $E(x|I) = \sum_{c \in C_G} \phi_c(X_c|I)$. The MAP method labels a random field of x^* that maximizes $P(x|I)$. For a fully connected pairwise CRF model, G is a complete graph defined on X, and C_G is the set of all unary and pairwise cliques. Therefore, the Gibbs free energy is expressed as:

$$E(x|I) = \sum_i \phi_u(x_i|I) + \sum_{i \neq j} \phi_p(x_i, x_j|I) \quad (2)$$

where i and j are the indices of pixels in I.

The unary term ϕ_u in Eqn. (2) is normally computed independently for each pixel, indicating the probability of each pixel belongs to each of the classes. The pairwise potential ϕ_p represents the penalty of assigning labels to pixel i and j at the same time. In fully connected CRF model, the pairwise cliques describe all two pairs of random variables. Subsequently, the mean field theory can be employed to produce an asymptotic solution.

In our implementation, we use the same pairwise cost ϕ_p as proposed by Krähenbühl et al. [13]. However, a different unary term ϕ_u is used, which is described in the next section. The pairwise cost consists of two terms that model the appearance and smoothness between pairs of pixels, expressed as:

$$\phi_p(x_i, x_j) = [x_i \neq x_j] g(i, j) \quad (3)$$

$$g(i,j) = \omega_1 \exp(-\frac{|p_i - p_j|^2}{2\theta_\alpha^2} - \frac{|I_i - I_j|^2}{2\theta_\beta^2}) + \omega_2 \exp(-\frac{|p_i - p_j|^2}{2\theta_\gamma^2}) \quad (4)$$

In equation (3), $[x_i \neq x_j]$ is an indicator function that indicates if the two labels at pixel i and j are the same. The first term (i.e. appearance term) in $g(i, j)$ encourages nearby pixels (determined by pixel location p) with similar intensities (denoted as I) to be the same class. The degrees of nearness and similarity are controlled by the parameters θ_α and θ_β respectively. The second term (i.e. smoothness) helps in removing small isolated regions that is controlled by θ_γ. ω_1 and ω_2 are the weights to balance the two terms. The parameters are determined experimentally using many different modalities of medical images and reported in section 3.1.

2.2. Unary Term for Interactive Image Segmentation

Unlike other machine learning based unary term modelling methods (e.g. [18] [19]), our method does not require multiple images of the same object and a pre-training step. In our proposed method, the unary term in Eqn. (2) is designed by considering both the distance and intensity similarity of a pixel to the scribbles annotated by the user, which is expressed as below.

$$\phi_u^c(i) = \lambda \exp(-0.5(\frac{I_i - m^c}{\sigma_1^c})^2) + (1-\lambda) \exp(-0.5(\frac{d_i^c}{\sigma_2})^2) \quad (5)$$

where m^c and σ_1^c are the mean and standard deviation of the intensity values annotated by the user for the c^{th} class respectively, which are calculated and updated during the user interaction process. The first term measures the likelihood of a pixel i belonging to class c, resulting in a probability map G_c. Fig. 1 (a) and (b) show an example image and some user annotations (yellow: foreground; white: background) respectively. Fig. 1 (c) and (d) are the intensity-based probability maps G_0 and G_1 for the background and the foreground

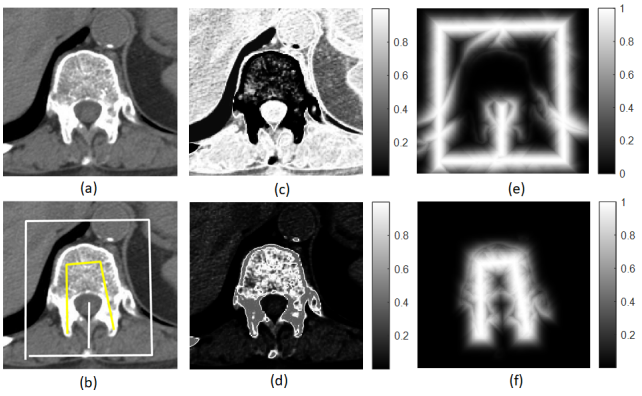

Figure 1: (a) An example image. (b) User annotation: white-background ($c = 0$); yellow-foreground ($c = 1$). (c) and (d) are the probability maps of background (G_0) and foreground (G_1) respectively. (e) and (f) are, respectively, the Gaussian weighted geodesic distance maps of background and foreground (second term in Eqn. (5)) with $\sigma_2 = 10$ pixels.

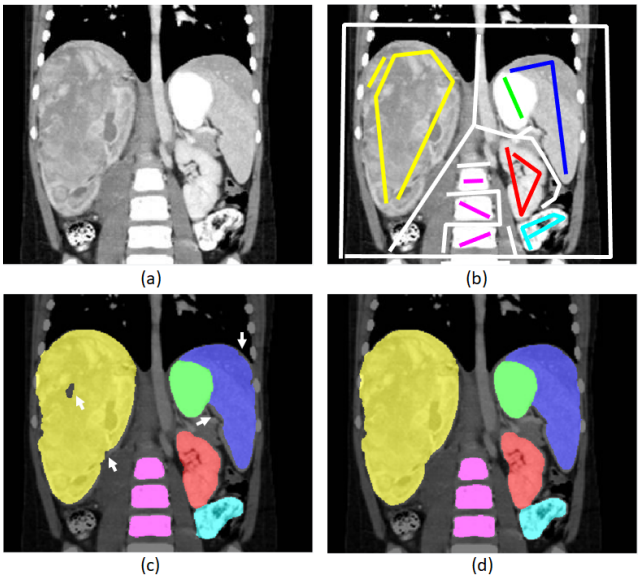

Figure 2: (a) An example image. (b) User annotation with multiple labels. (c) Segmentation result before refinement with some inaccurate regions indicated by white arrows. (d) Segmentation result after automatic refinement.

respectively. A brighter pixel indicates a higher probability of belonging to the corresponding class. d_i^c in Eqn. (5) is the minimum path between the i^{th} pixel to the nearest labelled pixel of the c^{th} class, which is calculated as geodesic distance. When calculating the minimum path, the locations along the path are weighted by the gradient of the probability image G_c. Therefore, the minimum path is the route that generates the smallest changes of G_c between two locations. The implementation is based on geodesic time algorithm [21]. The capture range of the distance measure is controlled by the parameter σ_2. Then the Gaussian weighted geodesic distance (second term in Eqn. (5)) is computed, as shown in Fig. 1 (e) (background) and Fig. 1 (f) (foreground) respectively. λ is used to balance the intensity term and the distance term. The parameter settings are discussed in section 3.1.

2.3. Image Segmentation and Refinement

The above constructed CRF can be solved by mean field theory. Krähenbühl et. al. [13] developed an iterative filter-based method for performing fast approximate maximum posterior marginal inference. The number of iterations is denoted as t. This fully connected CRF optimization method has not been applied to interactive image segmentation previously.

In the user annotation process, different class labels are required to be assigned to different objects of interest. The correct number of class labels is not required in the first annotation step and more class labels can be added at any stage of the user interaction. The annotation can be corrected/overwritten by new labels at the same location. For adjacent objects that share similar intensities, some scribbles of each class are expected to be drawn close to the shared boundary. For 3D volume annotation, annotations from different views are conducted separately but recorded in a single 3D volume. When a voxel is assigned by multiple labels from different views, the latest assigned label is used. For 3D images, the fully connected CRF optimization is performed in 3D.

Fig. 2 (a) shows an example image and Fig. 2 (b) is the scribbles that a user annotated. Note that these scribbles were interactively added based on intermediate segmentation results. The final result based on the annotations in Fig. 2 (b) is shown in Fig. 2 (c). As indicated by the white arrows in Fig. 2 (c), there are holes in certain regions and some inaccurately segmented regions along the boundaries of some organs. The user can keep adding more scribbles to these inaccurate regions until satisfied. However, it could be a tedious work to accurately refine these boundaries manually. Alternatively, we add an automatic segmentation refinement step to help the user in reducing the number of interactions.

The segmentation refinement is achieved as follows. (1) Replacing the probability map G_c (first term in Eqn. (5)) by the output probability map from the current CRF solution. (2) The geodesic distance map d^c in Eqn. (5) is then recalculated based on the new G_c. (3) The CRF optimization is applied again using the updated unary term, leading to filled holes and refined boundaries as shown in Fig. 2 (d). In our implementation, this refinement step runs as an extra step on every intermediate result, not only on the final result. It requires slightly longer computational time, but leads to fewer user interactions.

2.4. Entropy-based Slice Recommendation

For 3D image segmentation, the 3D volume normally contains many slices (typically more than one hundred) in the three different views (i.e. axial, coronal and sagittal). In our interactive image segmentation tool, the user is asked to start the annotation from the middle slice in each of the three views, which is more likely to contain the object of interest.

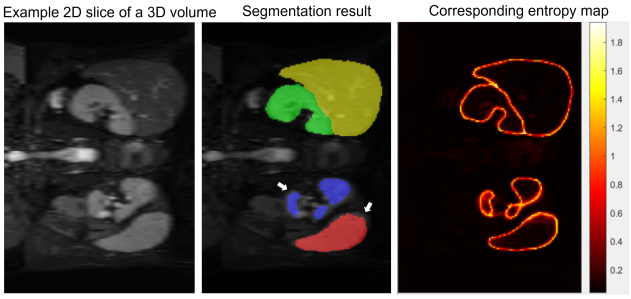

Figure 3: Images from left to right are an 2D slice of a 3D volume in the CHAOS dataset, the segmentation result at an intermediate iteration and the corresponding entropy map. White arrows indicate the image regions with larger segmentation error and corresponding to larger entropy values.

By only annotating one slice of any or all of the three views, the tool is able to generate an initial segmentation result. For other slices which are not annotated by the user, the closer to the annotated slice the more accurate the segmentation will be. Typically, the user needs to validate the segmentation result in each slice of all views and correct the results if not satisfied, which is a time-consuming process. In our software, the next slice in each of the three views that potentially contains the largest segmentation error is automatically suggested, based on the calculation of entropy. Entropy has been used as an indication of uncertainty by many research works (e.g Gal [22]). Hoebel et. al. have reported high correlations between the uncertainty measures and the corresponding Dice coefficient values [23]. In our case, the entropy H for the 3D volume is calculated as below.

$$H = -\sum_{c=0}^{K-1} G_c \log(G_c) \tag{6}$$

where G_c is the probability map for the c^{th} class generated by the CRF optimization. K is the total number of classes. Subsequently, the average entropy value for each slice in each of the three views is calculated. A larger entropy value indicates a higher uncertainty of the segmentation result. As the example shown in figure 3, higher entropy values are found at boundaries of the segmented objects that corresponding to larger segmentation errors (also see the highlighted regions by the white arrows in the middle image). The slice that produces the largest average entropy value for each view is suggested to the user for further annotation. In our experiments, we have found that this slice recommendation function significantly improved the annotation efficiency, especially in the first few interactive actions. This function is an optional feature to the user, where a button in the GUI has to be clicked every time of requiring a suggestion. Note that the user still needs to validate the segmentation result on every slice in each view to ensure a high quality segmentation output.

Table 1
Hyper parameter setting.

Symbol	Meaning	Value
θ_α	Nearness controller for the appearance kernel (Eqn.(4))	20 (2D & 3D)
θ_β	Similarity controller for the appearance kernel (Eqn.(4))	1 (2D & 3D)
θ_γ	Controller for the smoothness kernel (Eqn. (4))	1 (2D & 3D)
ω_1	Weight of the Gaussian appearance kernel (Eqn. (4))	1 (2D & 3D)
ω_2	Weight of the smoothness kernel (Eqn. (4))	5 (2D & 3D)
σ_2	Distance measure controller (Eqn. (5))	4 (2D & 3D)
λ	Weight for balancing the intensity term and the distance term (Eqn. (5))	0.1 (2D & 3D)
t	Number of iterations in CRF optimization	10 (2D);3 (3D)

3. Parameter Settings and Graphical User Interface

3.1. Parameter Settings

The meaning and values for all the hyper parameters described in section 2 are listed in Table 1. The parameter values were determined by evaluating the software on various 2D and 3D images described in section 4.1. In the context of medical image segmentation, the parameters were optimized in favor of responding to the user's annotations to improve the segmentation accuracy rather than minimizing the number of user's interactions. Hence we used relatively smaller values for σ_2 and λ in Eqn. (5) to make the tool more responsive to the user's annotations. After tuning the parameters, the values were all fixed for all testing experiments reported in this paper. As shown in the graphical user interface in Fig. 4, our software does not require the user to adjust any parameters.

3.2. Graphical User Interface

The software was implemented in Matlab (version 2020b) and compiled to an executable file (.exe). The core functions of CRF optimisation were written in C++. Currently, it only supports Windows operation system and has been tested on Windows 10. The graphical user interface is shown in Fig. 4. The basic functions labeled in the figure are briefly described as follows. (1) Load a 2D image to be segmented with the supporting file formats of .jpg, .tiff, .bmp, .png, DICOM and .mat. (2) Load a 3D volume (or multiple 2D slices) with the supporting formats of .mat, DICOM and .nii. (3) Load segmentation result in .mat or .nii format. (4) Resample the 3D volume into isotropic physical unit (mm). (5) Perform CRF segmentation after user annotation. (6) Enable/disable overlapping the segmented results to the original image. (7) Automatically suggest the best slice to be annotated for each of the views in 3D. (8) Display the measured areas (2D) or volume (3D) in physical unit (if unit is known) for each of

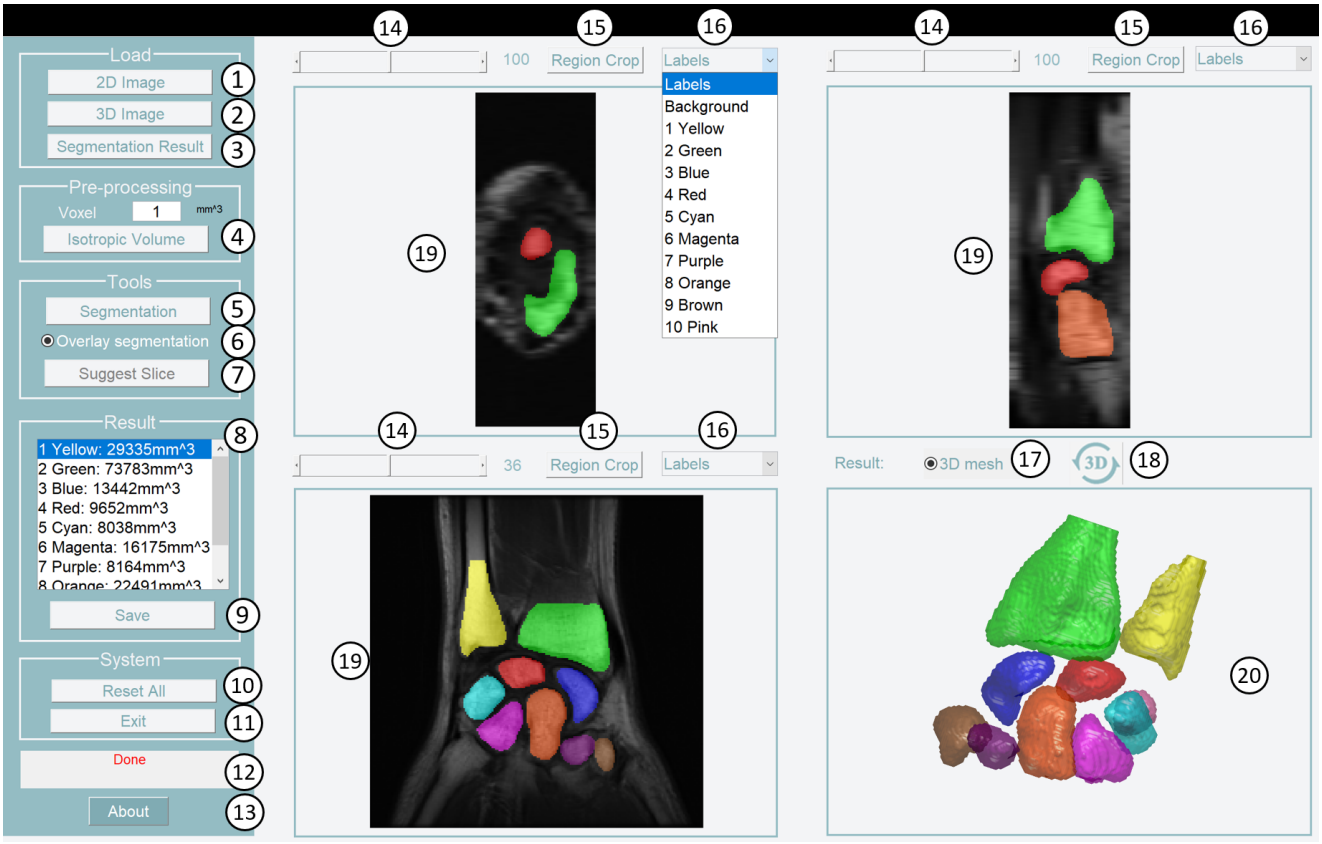

Figure 4: Graphical user interface of the developed software. An example 3D wrist MRI, in which 10 bones were segmented by our software as indicated by different colours.

the segmented classes. (9) Save the segmented result in the supported formats: 2D (.mat, .png, .bmp and .tiff) and 3D (.mat and .nii). (10) Clear all views and data. Reset all parameters to the default settings. (11) Exit the software and clear memory. (12) Information bar that indicates the current status of calculation. (13) Brief information about the software and user instructions. (14) Slide bars to change the slices in the corresponding views for 3D volumes. (15) Crop a smaller region of interest from the input 2D or 3D image in the corresponding view. (16) A drop down list of labels in the corresponding view for user annotation. Currently it supports up to 10 foreground classes plus the background. (17) Enable/disable 3D mesh view of the segmented result for a 3D volume. (18) Tool for manipulation of the 3D mesh model. (19) Image viewer for displaying and annotating 2D and 3D images. 2D image is displayed in the top left viewing window. (20) Viewing window for displaying the 3D mesh model of the segmentation result for a 3D volume.

4. Evaluation Results

4.1. Material

Combined (CT-MR) Healthy Abdominal Organ Segmentation dataset (CHAOS) [24]: The CHAOS challenge aims to segment liver, kidneys and spleen in the abdominal region in CT and MRI data. The manual annotation process was produced by two teams, both of which include a radiology expert and an experienced medical image processing scientist. After the manual annotation of both teams, a third radiology expert and another medical imaging scientist analyzed the labels, which were fine-tuned according to the discussions between annotators and controllers. The CHAOS dataset has high quality ground truth segmentation masks, hence selected to evaluate the segmentation accuracy of our tool. The CT dataset was acquired from 40 different patients and only has the segmentation mask of liver. The MRI dataset contains two sequences (i.e. T1- DUAL and T2-SPIR) of 40 patients from 1.5T MRI scanners. The segmentation of T2-SPIR MRI dataset contains liver, both kidneys and spleen, which is a more challenging multi-label segmentation task, hence selected as the dataset to evaluate our method.

Wrist CT dataset: this wrist CT dataset was used in our previous work [25] that contains CT image from 25 subjects. Each subject was imaged at five different wrist poses: neutral and four extreme poses in radial-ulnar and flexion-extension. The pixel spacing is $0.29mm \times 0.29mm$ with a slice thickness of $0.625mm$. It is a challenging image segmentation task, as 10 bones (i.e. ulnar, radius and eight carpal bones) are required to be segmented in a small wrist region of the CT image. All the bones have similar intensity values and in close contact with each other.

For the purpose of parameter tuning as described in section 3.1, a variety of medical images that cover different

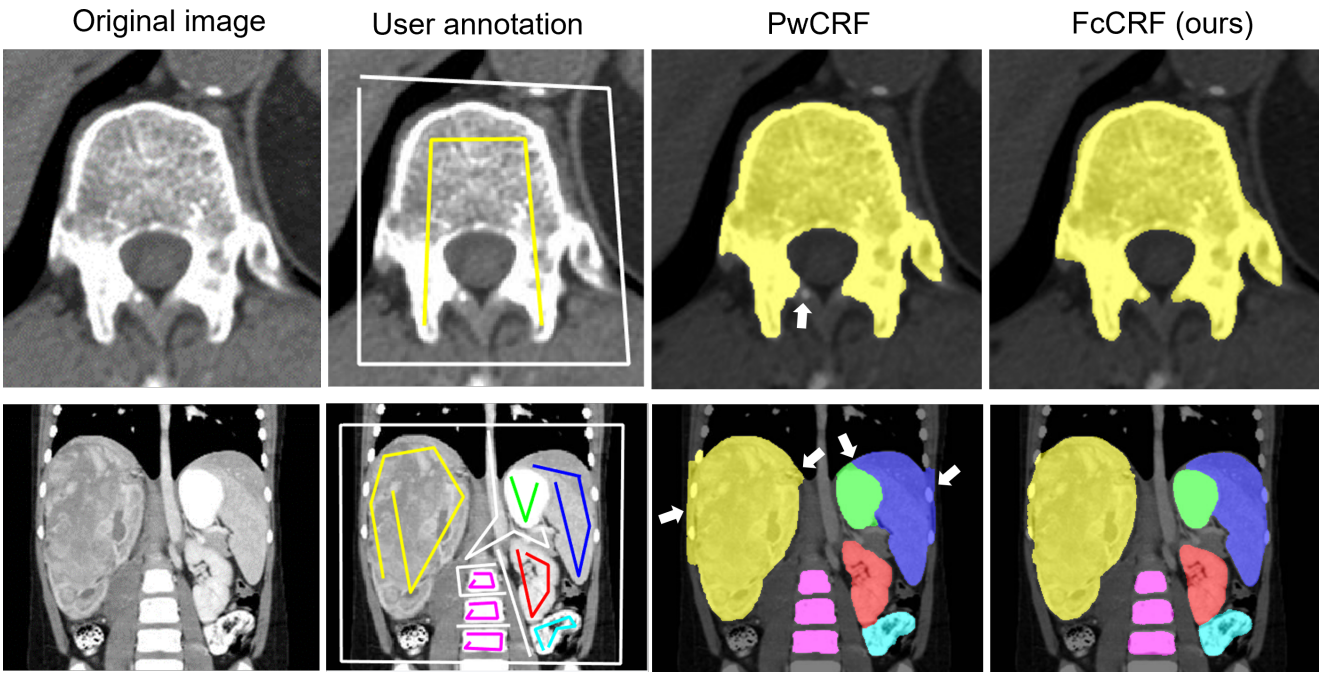

Figure 5: Comparison of local pair-wise CRF (PwCRF) and fully-connected CRF (FcCRF). Top row: an example of binary class segmentation. Bottom row: an example of multiple class segmentation.White arrows indicate inaccurate segmentation locations.

Table 2
Computational time using local pair-wise CRF (PwCRF) and fully-connected CRF (FcCRF) for the examples in Figure 4."Total" indicates the total computational time including unary term calculation. "CRF optimization" only includes the CRF optimzation process.

Image	PwCRF	FcCRF
Binary class	5.11 s (total); 0.07 s (CRF optimization)	0.31 s (total); 0.10 s (CRF optimization)
Multiple class	15.89 s (total); 2.24 s (CRF optimizaion)	1.12 s (total); 0.36 s (CRF optimization)

imaging modalities and different organs were obtained from a number of local and public datasets, such as plain wrist X-ray image from Chen et al. [26], contrast enhanced breast MRI from [27], ultrasound breast image from Al-Dhabyani et al. [28], retinal images from Budai et al. [29]. These images were used as a validation set for parameter tuning of our method. Some qualitative segmentation results of these images are presented in section 4.5.

4.2. Comparison of Local Pair-wise CRF and Fully-connected CRF

One of the most widely used CRF based optimisation that has been applied to interactive image segmentation is based on the pair-wise potential of nearest neighbors [9] (denoted as PwCRF). We compare the performance of PwCRF and the fully-connected CRF approximation (denoted as FcCRF) in terms of qualitative segmentation accuracy and computational time using the same user annotation.

The PwCRF implementation is the baseline method implemented by Kohli et al. [30]. The core PwCRF optimization using α expansion was implemented in C++ based on the paper [9]. The core FcCRF optimization using mean field approximation was adapted from the C++ implementation by Kamnitsas et al. [31]. The evaluation was performed on the same machine.

We performed the comparison based on a binary class segmentation and a multi-class segmentation. Figure 5 show the original input images, user annotations and segmentation results using PwCRF and FcCRF respectively. It can be seen that the segmentation result of using PwCRF, especially for the multiple class case, is less accurate at the boundaries of the objects than the FcCRF method (highlighted by white arrows). More importantly, in the context of interactive segmentation, the FcCRF runs much faster than the PwCRF, as shown in Table 2. As listed in Table 2, the total computational time after user annotation til obtaining the segmentation result, FcCRF only took 0.31 s and 1.12 s for the binary and multi-class case respectively. It is significantly quicker than the PwCRF method (5.11s and 15.89 s respectively). This computational difference is partly due to the unary term calculation. The PwCRF implementation by Kohli et al. [30] used K-means clustering to calculate the unary term, while our FcCRF method is a simple Gaussian-based unary term calculation. Our FcCRF method generated more accurate segmentation result, as shown in Figure 5. This demonstrates the advantage of our proposed unary

Table 3
Segmentation accuracy of chaos dataset for liver, kidney and spleen segmentation. Mean ± standard deviation values of Dice Coefficient (DC) and Average Symmetric Surface Distance (ASSD) are reported.

Metrics	Liver	Left kidney	Right kidney	Spleen
DC	0.923 ± 0.002	0.906 ± 0.023	0.894 ± 0.018	0.875 ± 0.019
ASSD(mm)	2.504 ± 0.080	1.616 ± 0.388	1.752 ± 0.298	1.623 ± 0.248

term compared with an existing popular method. If only the CRF optimization process is considered, the FcCRF method is slightly slower than PwCRF in the binary case (0.1 s and 0.07 s respectively), but 6 times faster for the multi-class case. The computational difference becomes more significant in 3D case.

It is worth noting that, for both PwCRF and FcCRF, the optimisation time at each iteration remain roughly constant. Hence the total computational time of both methods are proportional to the number of iterations given the same input image. Hence the total run time of FcCRF is much shorter than the PwCRF, due to more accurate segmentation results at each iteration (therefore fewer number of interactions) and shorter optimisation time at each iteration. This is the main reason that we selected FcCRF in our interactive image segmentation software.

4.3. Evaluation on Segmentation Accuracy

Five T2-SPIR MRIs from the CHAOS dataset were randomly selected for evaluating the segmentation accuracy of a multi-label segmentation task in 3D image. Dice coefficient (DC) and average symmetric surface distance (ASSD) were used as the evaluation metrics. The DC is a widely used measurement in image segmentation evaluation, which indicates the volume agreement between the generated segmentation result and the ground truth segmentation mask (i.e. 0 and 1 indicate the worst and best segmentation results respectively). The ASSD determines the average difference between the surface of the segmented object and the ground truth segmentation mask in 3D. After the border voxels of the segmentation output and the ground truth mask are determined, those voxels that have at least one neighbor from a predefined neighborhood that does not belong to the object are collected. For each collected voxel, the closest voxel in the other set is determined and the average of all these distances derive the ASSD measure (0*mm* for a perfect segmentation, max distance of the image for the worst case). The mean ± standard deviation values of DC and ASSD for the five MRIs are reported in Table 3.

For interactive image segmentation, the segmentation result is highly dependent on the number of interactions and experience of the annotator. The results presented in Table 3 were produced by an annotator without medical background. In our proposed method, the volume was firstly cropped and re-sampled to an isotropic volume, and the segmentation was then performed in 3D. The size of the volume was approximately $120 \times 170 \times 120$ (the sizes of different volumes are slightly different) with the voxel size of $2mm^3$. The software ran on a laptop with an Intel i5-6300U 2.4 GHz processor and 8 GB memory. The total segmentation time is highly dependent on the size of the volume and the number of interactions required. The average number of interactions and time in these experiments were approximately 20 and 25 minutes per volume respectively. The segmentation accuracy is comparable to the results reported in the literature [32]. Fig. 6 shows the visual result of an example that produced the lowest mean DC value of the four organs (liver: 0.88; both kidneys: 0.86; spleen: 0.83) in the five segmented volumes. From the 2D slices in Fig. 6, it can be seen that the segmentation result using our software (green contours) is agreed with the ground truth annotation (red contours) in most regions, except for the peripheral regions of some organs (e.g. pelvis of kidney). This disagreement is highly related to the experience and knowledge of the annotator. Technically, higher segmentation accuracy can be achieved by more user interactions to refine these regions. From the mesh models shown in Fig. 6, it can be seen that the ground truth annotation is performed in a multiple 2D slice manner, while our method is performed in 3D, which may also lead to the result discrepancy.

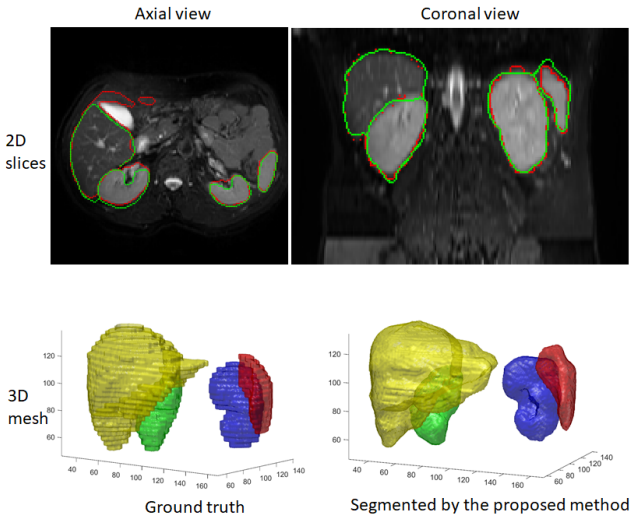

Figure 6: Top row: Visual segmentation result in axial view and coronal view (red: ground truth; green: the proposed method). Bottom row: visual segmentation results in 3D mesh model of ground truth and segmentation result using the proposal method (liver: yellow; kidneys: green & blue; spleen: red).

4.4. Evaluation on Repeatability and Reliability

To evaluate the repeatability of the interactive image segmentation tool, three annotators performed image segmenta-

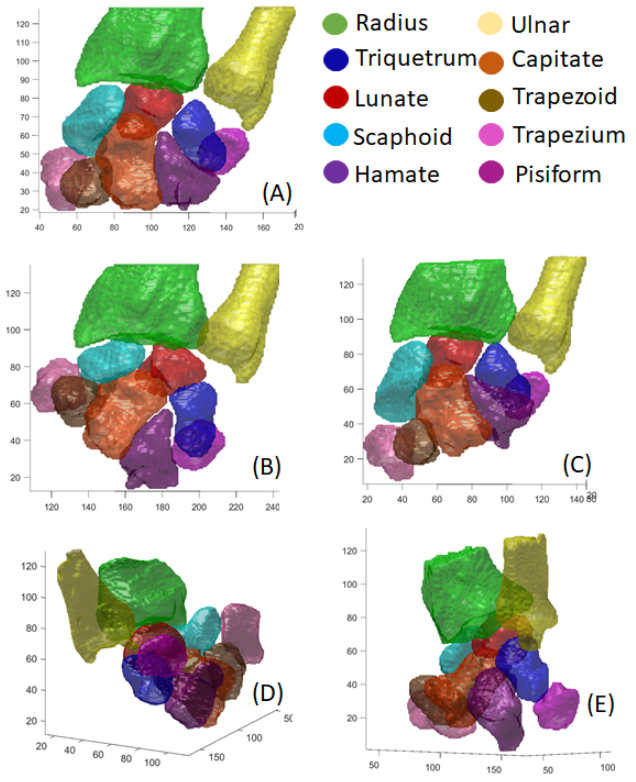

Figure 7: Segmentation results of carpal bones in CT volumes of different wrist poses from the same subject. (A) Neutral (B) Radial-deviation (C) Ulnar-deviation (D) Flexion (E) Extension.

Table 4
Percentage of standard deviation for volume of carpal bones segmented from different poses of the same subject. The carpal bones are: Triquetrum (Tri), Lunate (Lun), Scaphoid (Sca), Pisiform (Pis), Hamate (Ham), Capitate(Cap), Trapezoid (Trd) and Trapezium (Trm).

Bones	Tri	Lun	Sca	Pis	Ham	Cap	Trd	Trm
Std%	2.69	3.05	2.99	2.81	2.11	3.15	1.41	2.41

tion on the same set of three T2-SPIR MRIs from the CHAOS dataset. Prior to the experiments, the three annotators were briefly trained by showing the organs of interest and the corresponding reference annotation of an independent T2-SPIR image. This helps to minimize the effects of knowledge discrepancy between the annotators. The annotators were given sufficient time to complete the segmentation task until they were satisfied with the segmentation result. Subsequently, the intraclass correlation coefficient (ICC) [33] was calculated based on the DC values to measure the performance consistency of different annotators. The average ICC score for all class labels is 0.7618, which indicates a good agreement (follow the guideline by Koo et al. [34]) between the segmentation results of different annotators using our software.

One of the key aims of medical image segmentation is the quantitative measurement of the region of interest, such as volumes of organs or tumors. The reliability of the measured volume is crucial for downstream clinical decision making tasks. Here, we use the wrist CT dataset to assess the reliability of the segmentation result produced by our software. A single annotator performed segmentation of eight carpal bones using our software on the CT images of three subjects, each contains CT volumes that were captured at five different wrist poses. The ulnar and radius bones were also segmented as the reference bones but were not considered for the performance evaluation. The assumption is that the measured volumes of the carpal bones at different wrist poses of the same subject should be the same. Fig. 7 shows the results of the carpal bone segmentation in different wrist poses of the same subject using our software. The variations of the measured bone volumes are listed in Table 4. The Std% value in Table 4 is calculated by using the standard deviation of the bone volumes measured from different poses divided by the corresponding mean bone volumes, and then averaged across the three subjects. An average of 2-3% of volume variations in our segmentation results indicates a small error range in measuring the bone volumes. The ICC was also calculated based on the segmented bone volumes to measure the consistency of the measurements across different poses. An ICC value of 0.9769 is achieved, which indicates a high consistency of the bone volumes measured in different wrist poses.

4.5. Evaluation on Efficiency

The high computational efficiency of the fully-connected CRF solution used in our method has also been demonstrated and compared to other CRF optimization methods by Krähenbühl et al. [13].

First, we demonstrate the efficiency of the proposed slice recommendation function. One randomly selected T2-SPIR MRI in the CHAOS dataset was used in this experiment. Two annotators were asked to segment four organs (liver, two kidneys and spleen) from the same 3D volume with a given initial annotation. Each of the annotators performed twice on the segmentation task, one with and one without using the slice recommendation function. "With the slice recommendation" was performed first, hence the result of "without slice recommendation" could be slightly better than reality due to the previous familiarization with the data. Despite this potential bias, the segmentation quality measured by Dice coefficient (average of the four organs) from both annotators increased much faster by using the slice recommendation function, especially in the first few user interactions (shown in figure 8). This demonstrates the improved segmentation efficiency by using the proposed slice recommendation function.

Next, we demonstrate the segmentation efficiency of our software in comparison with a widely used manual segmentation tool (i.e. 3D Slicer [2]) by drawing polynomial lines in a slice by slice manner. A single annotator performed organ segmentation using 3D Slicer and our software on a T2-SPIR MRI in the CHAOS dataset. The DC value using 3D slicer for liver, left kidney, right kidney, and spleen

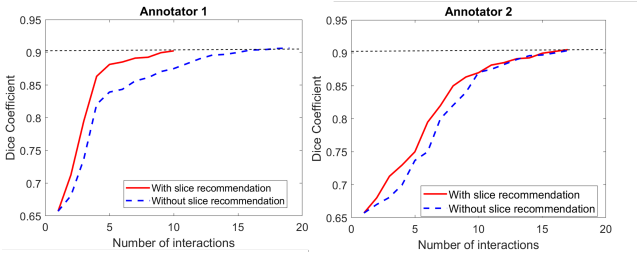

Figure 8: Comparison of segmentation quality (Dice coefficient) with/without slice recommendation from two annotators.

were 0.91, 0.91, 0.89 and 0.88, which achieved similar performance to our method (refer to Table 3). However, the time used in 3D slicer was 50 minutes, which was significantly higher than the time used in our method (average of 25 minutes).

It is even more challenging in segmenting the wrist CT data. Due to multiple carpal bones having similar shapes and intensities, it is extremely difficult to maintain the track of the class labels of different bones during the slice by slice manual segmentation process. Using 3D slicer, the annotator made a lot of efforts in checking the slice-wise context to ensure consistent class labels across different slices. In contrast, our method is performed in 3D and the whole 3D volume is labeled by only annotating a few slices in different views. For the wrist CT data, 3D slicer required about 90 minutes to label a single volume, and the annotator needs to have a good knowledge about the anatomy of the wrist. Our software only took about 30 minutes to segment the carpal bones. Especially, our software does not require accurate contour tracing which needs lower concentration level from the annotator.

4.6. Gallery of Segmentation Results

Besides the above systematic evaluations, we also show some qualitative segmentation results of using our software on a variety of medical images in Fig. 9, including plain X-ray images for bone segmentation, 3D contrast enhanced breast MRI for tumor segmentation, breast ultrasound image for tumor segmentation and retinal image for blood vessel, macula and optical disc segmentation. It can be observed from Fig. 9 that the software works well for segmenting organs, tumors in any of the given medical modalities, but failed to segment part of the blood vessels in the retinal image. This type of thin linear structure, which distributed across the whole image, requires the user annotations to cover the whole image. In this case, it makes the user annotation extremely challenging and time consuming using our method. In this example, the user only annotated a very small portion of the image leading to an unsatisfactory result for part of the blood vessels. Hence, alternative solutions are strongly recommended if such a linear structure needs to be segmented. For example, a deep learning method that handles inaccurate annotations were proposed by Zhang et al. [35] for segmenting linear structures.

5. Discussion and Conclusions

In this paper, we have proposed to use an efficient fully connected CRF optimizer to achieve multi-class 2D and 3D medical image segmentation. Based on the CRF optimizer, we have developed an interactive image segmentation tool for medical image analysis. Our tool does not require parameter tuning for different image modalities and dimensions. It is also featured with a slice recommendation function to achieve efficient user interactions for 3D images. The method has been comprehensively evaluated in terms of segmentation accuracy, repeatability, reliability and efficiency on different medical imaging datasets and applications. Our method performs well on the segmentation of regular shaped objects (e.g. organs, bones, tumors, etc.), but it is less efficient in segmenting thin linear structures, such as blood vessels in retinal image. The software is freely available for research purposes. In future work, we intend to incorporate a robust feature learning process to further reduce the number of user interactions and maintain a high segmentation accuracy.

References

[1] P. A. Yushkevich, J. Piven, H. Cody Hazlett, R. Gimpel Smith, S. Ho, J. C. Gee, G. Gerig, User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability, Neuroimage 31 (2006) 1116–1128.

[2] A. Fedorov, R. Beichel, J. Kalpathy-Cramer, J. Finet, J.-C. Fillion-Robin, S. Pujol, C. Bauer, D. Jennings, F. Fennessy, M. Sonka, et al., 3d slicer as an image computing platform for the quantitative imaging network, Magnetic resonance imaging 30 (2012) 1323–1341.

[3] N. Sharma, L. M. Aggarwal, Automated medical image segmentation techniques, Journal of medical physics/Association of Medical Physicists of India 35 (2010) 3.

[4] M. H. Hesamian, W. Jia, X. He, P. Kennedy, Deep learning techniques for medical image segmentation: achievements and challenges, Journal of digital imaging 32 (2019) 582–596.

[5] N. R. Pal, S. K. Pal, A review on image segmentation techniques, Pattern recognition 26 (1993) 1277–1294.

[6] M. Kass, A. Witkin, D. Terzopoulos, Snakes: Active contour models, International journal of computer vision 1 (1988) 321–331.

[7] T. Chan, L. Vese, An active contour model without edges, in: International Conference on Scale-Space Theories in Computer Vision, Springer, 1999, pp. 141–151.

[8] V. Vezhnevets, V. Konouchine, Growcut: Interactive multi-label nd image segmentation by cellular automata, in: proc. of Graphicon, volume 1, Citeseer, 2005, pp. 150–156.

[9] Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (2001) 1222–1239.

[10] J. Besag, On the statistical analysis of dirty pictures, Journal of the Royal Statistical Society: Series B (Methodological) 48 (1986) 259–279.

[11] J. S. Yedidia, W. T. Freeman, Y. Weiss, Constructing free-energy approximations and generalized belief propagation algorithms, IEEE Transactions on information theory 51 (2005) 2282–2312.

[12] Y. Boykov, V. Kolmogorov, An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision, IEEE transactions on pattern analysis and machine intelligence 26 (2004) 1124–1137.

[13] P. Krähenbühl, V. Koltun, Efficient inference in fully connected crfs with gaussian edge potentials, Advances in neural information processing systems 24 (2011) 109–117.

[14] C. Rother, V. Kolmogorov, A. Blake, " grabcut" interactive fore-

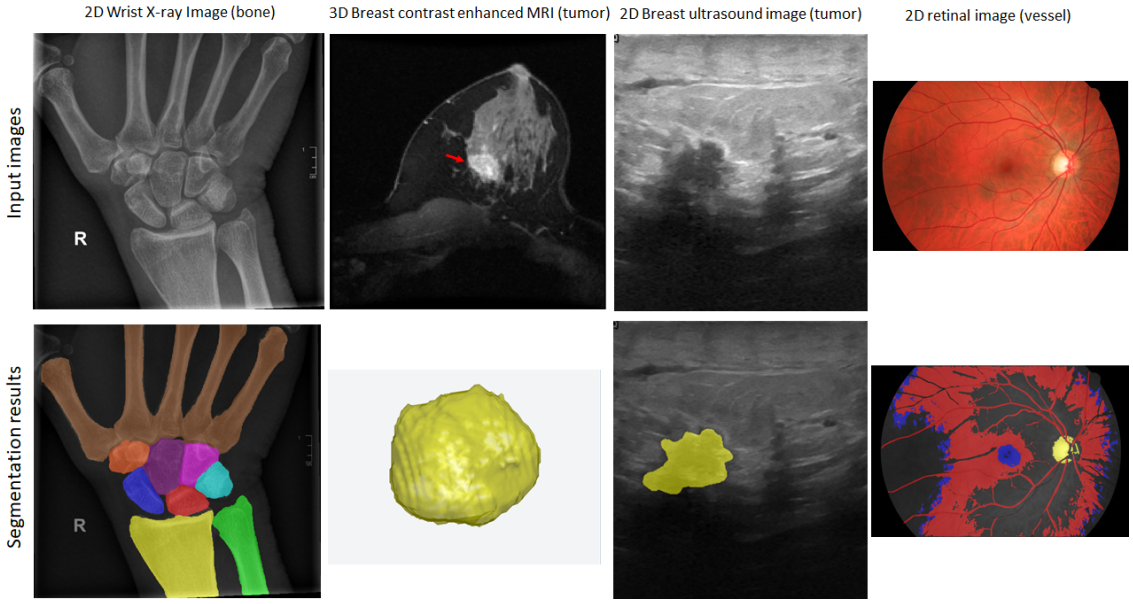

Figure 9: Qualitative segmentation results of different medical images and applications. The bones, organs and tumors can be efficiently segmented by our method. For the retinal image segmentation, it requires tremendous user annotation efforts to segment the linear structures in the whole image. Hence, it is not recommend to use our method in segmenting thin linear structures.

ground extraction using iterated graph cuts, ACM transactions on graphics (TOG) 23 (2004) 309–314.

[15] P. Kohli, M. P. Kumar, P. H. Torr, P3 & beyond: Solving energies with higher order cliques, in: 2007 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2007, pp. 1–8.

[16] V. Vineet, J. Warrell, P. Sturgess, P. H. Torr, Improved initialization and gaussian mixture pairwise terms for dense random fields with mean-field inference., in: BMVC, 2012, pp. 1–11.

[17] M. Rajchl, M. C. Lee, O. Oktay, K. Kamnitsas, J. Passerat-Palmbach, W. Bai, M. Damodaram, M. A. Rutherford, J. V. Hajnal, B. Kainz, et al., Deepcut: Object segmentation from bounding box annotations using convolutional neural networks, IEEE transactions on medical imaging 36 (2016) 674–683.

[18] D. Lin, J. Dai, J. Jia, K. He, J. Sun, Scribblesup: Scribble-supervised convolutional networks for semantic segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 3159–3167.

[19] G. Wang, W. Li, M. A. Zuluaga, R. Pratt, P. A. Patel, M. Aertsen, T. Doel, A. L. David, J. Deprest, S. Ourselin, et al., Interactive medical image segmentation using deep learning with image-specific fine tuning, IEEE transactions on medical imaging 37 (2018) 1562–1573.

[20] X. Chen, J. Graham, C. Hutchinson, Integrated frameworkfor simultaneous segmentation and registration of carpal bones, in: 2011 18th IEEE International Conference on Image Processing, IEEE, 2011, pp. 433–436.

[21] P. Soille, Generalized geodesy via geodesic time, Pattern Recognition Letters 15 (1994) 1235–1240.

[22] Y. Gal, Uncertainty in Deep Learning, Ph.D. thesis, University of Cambridge, 2016.

[23] K. Hoebel, V. Andrearczyk, A. Beers, J. Patel, K. Chang, A. Depeursinge, H. Müller, J. Kalpathy-Cramer, An exploration of uncertainty information for segmentation quality assessment., in: SPIE Medical Imaging, 2020, p. 11313.

[24] A. E. Kavur, M. A. Selver, O. Dicle, M. Barış, N. S. Gezer, CHAOS - Combined (CT-MR) Healthy Abdominal Organ Segmentation Challenge Data, 2019. URL: https://doi.org/10.5281/zenodo.3362844. doi:10.5281/zenodo.3362844.

[25] X. Chen, J. Graham, C. Hutchinson, L. Muir, Automatic generation of statistical pose and shape models for articulated joints, IEEE transactions on medical imaging 33 (2013) 372–383.

[26] X. Chen, J. Graham, C. Hutchinson, L. Muir, Automatic inference and measurement of 3d carpal bone kinematics from single view fluoroscopic sequences, IEEE transactions on medical imaging 32 (2012) 317–328.

[27] o. b. o. t. I.-S. . N. David Newitt, Nola Hylton, A. . T. Team, Multicenter breast dce-mri data and segmentations from patients in the i-spy 1/acrin 6657 trials, 2016. URL: https://doi.org/10.7937/K9/TCIA.2016.HdHpgJLK.

[28] W. Al-Dhabyani, M. Gomaa, H. Khaled, A. Fahmy, Dataset of breast ultrasound images, Data in brief 28 (2020) 104863.

[29] A. Budai, R. Bock, A. Maier, J. Hornegger, G. Michelson, Robust vessel segmentation in fundus images, International journal of biomedical imaging 2013 (2013).

[30] P. Kohli, A. Osokin, S. Jegelka, A principled deep random field model for image segmentation, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition, 2013. doi:10.1109/CVPR.2013.257.

[31] K. Kamnitsas, C. Ledig, V. F. Newcombe, J. P. Simpson, A. D. Kane, D. K. Menon, D. Rueckert, B. Glocker, Efficient multi-scale 3d cnn with fully connected crf for accurate brain lesion segmentation, Medical image analysis 36 (2017) 61–78.

[32] A. E. Kavur, N. S. Gezer, M. Barış, Y. Şahin, S. Özkan, B. Baydar, U. Yüksel, Ç. Kılıkçıer, Ş. Olut, G. B. Akar, et al., Comparison of semi-automatic and deep learning-based automatic methods for liver segmentation in living liver transplant donors, Diagnostic and Interventional Radiology 26 (2020) 11.

[33] K. O. McGraw, S. P. Wong, Forming inferences about some intraclass correlation coefficients., Psychological methods 1 (1996) 30.

[34] T. K. Koo, M. Y. Li, A guideline of selecting and reporting intraclass correlation coefficients for reliability research, Journal of chiropractic medicine 15 (2016) 155–163.

[35] N. Zhang, S. Francis, R. A. Malik, X. Chen, A spatially constrained deep convolutional neural network for nerve fiber segmentation in corneal confocal microscopic images using inaccurate annotations, in: 2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI), IEEE, 2020, pp. 456–460.