Learning multi-level structural information for small organ segmentation

Yueyun Liu[a], Yuping Duan[a,*], Tieyong Zeng[b]

[a] Center for Applied Mathematics, Tianjin University, Tianjin, 300072, China
[b] Department of mathematics, The Chinese University of Hong Kong, Shatin, NT, Hong Kong

A B S T R A C T

Deep neural networks have achieved great success in medical image segmentation problems such as liver, kidney, the accuracy of which already exceeds the human level. However, small organ segmentation (e.g., pancreas) is still a challenging task. To tackle such problems, extracting and aggregating multi-scale robust features become essentially important. In this paper, we develop a multi-level structural loss by integrating the region, boundary, and pixel-wise information to supervise feature fusion and precise segmentation. The novel pixel-wise term can provide information complementary to the region and boundary loss, which helps to discover more local information from the image. We further develop a multi-branch network with a saliency guidance module to better aggregate the three levels of features. The coarse-to-fine segmentation architecture is adopted to use the prediction on the coarse stage to obtain the bounding box for the fine stage. Comprehensive evaluations are performed on three benchmark datasets, i.e., the NIH pancreas, ISICDM pancreas, and MSD spleen dataset, showing that our models can achieve significant increases in segmentation accuracy compared to several state-of-the-art pancreas and spleen segmentation methods. Furthermore, the ablation study demonstrates the multi-level structural features help both the training stability and the convergence of the coarse-to-fine approach.

1. Introduction

Image segmentation is a central theme in medical image processing, which separates image volume into sub-regions according to biological structure and function. Various methods have been developed for image segmentation such as region growing[1], clustering [2], graph cuts method [3] and model-based [4,5]. However, segmenting a small organ from CT scans is still challenging due to the high variabilities in shape, size, and location. The pancreas is a typical representative of small organs in the human body. The traditional multi-atlas registration-based segmentation [6] can achieve the segmentation accuracy on the liver, kidneys, and spleen over 90%, while the segmentation accuracy on the pancreas is only around 70%. Thanks to the great progress in deep learning, especially convolutional neural networks (CNN) such as FCN [7], U-Net [8] and DeepLab [9], the accuracy of medical image segmentation has been improved significantly during the last several years. However, it remains difficult to precisely segment boundaries due to ambiguity in discriminating pixels around boundaries. Thus, multi-scale features [10,11] and efficient feature fusion methods [11–13] are intensively studied to assist precise segmentation. Xie and Tu [10] developed an end-to-end edge detection system to automatically learn the type of rich hierarchical features. Chen *et al.* [14] proposed an efficient deep contour-aware network for accurate gland segmentation under a unified multi-task learning framework. Shen *et al.* [15] proposed a multi-task full convolutional network architecture to jointly learn to predict tumor regions and tumor boundaries. Xu *et al.* [16] designed a deep multichannel framework to automatically exploit and fuse complex information including regional, location, and boundary cues. Duan *et al.* [17] estimated the probability maps over the region and edge locations using a fully convolutional network, which was incorporated in a single nested level set optimization framework to achieve multi-region segmentation. Pang *et al.* [18] introduced the boundary attention module to bridge the semantic gap between multi-level features. Zhang and Pang [19] identified the edge and saliency information for segmentation and presented the cross-guidance network. Deep edge priors have also been introduced into the networks to precisely integrate the edge information for dealing with image denoising, super-resolution and segmentation tasks [20–22]. Recently, Zhou *et al.* [23] developed a novel multi-label learning network for RGB-thermal urban scene semantic seg-

E-mail addresses: yueyunliu@tju.edu.cn (Y. Liu), yuping.duan@tju.edu.cn (Y. Duan), zeng@math.cuhk.edu.hk (T. Zeng).

mentation, which trained the network in terms of semantic, binary, and boundary characteristics. Zhou *et al.* [24] proposed a crossflow and cross-scale adaptive fusion network, and used the purification loss to precisely learn the boundaries and details of the objects. Besides, regularization methods have been introduced into CNNs to make use of prior information of image edges such as total variation regularization [25] and graph total variation regularization [26], which was integrated into the architecture of CNNs through the softmax activation functions. The aforementioned approaches improve the segmentation accuracy by either adopting multiple task fashion or introducing extra network structures to optimize the context feature extraction process. Nevertheless, the design of the loss function used to measure the similarities between the predictions and ground truths also plays an important role in precise segmentation.

Suppose a 2D image $\mathrm{U} \in X$, where the intensity at a specified position is denoted as $\mathrm{U}(x, y)$ and the label data $\mathrm{V} \in X$ shares the same dimension with U with X being some topological vector space. Then the feed-forward CNNs trained for image segmentation tasks can be formally expressed as the following bi-level optimization problem

$$\begin{aligned} \min_{\theta} \quad & E(\mathrm{P}_\theta, \mathrm{V}), \\ \text{s.t.} \quad & \mathrm{P}_\theta = \arg\min_{\mathrm{P} \subset X} \mathcal{F}(\mathrm{P}; \mathrm{U}, \theta), \end{aligned} \tag{1}$$

where the upper-level minimization is known as the loss function $E : X \times X \to \mathbb{R}$ mapping real-valued variables into real numbers, and the lower-level minimization denotes a CNN model $\mathcal{F}(\cdot; \mathrm{U}, \theta)$ with P_θ as its output. The loss function is to measure the difference between P_θ, which is the minimizer of $\mathcal{F}(\cdot; \mathrm{U}, \theta)$, and the labeled ground truth, which has a similar function as the objective functional in variational models. The active contour (AC) model proposed by Chan and Vese [5] has achieved great success in foreground-background segmentation by deploying efficient priors on image boundary, to minimize

$$\begin{aligned} E_{AC}(c_1, c_2, \Gamma) = {} & \mu \cdot \mathrm{Length}(\Gamma) + \nu \cdot \mathrm{Area}(inside(\Gamma)) \\ & + \lambda_1 \int_{inside(\Gamma)} |\mathrm{U}(x, y) - c_1|^2 dxdy \\ & + \lambda_2 \int_{outside(\Gamma)} |\mathrm{U}(x, y) - c_2|^2 dxdy, \end{aligned} \tag{2}$$

where Γ is a closed curve, c_1, c_2 are the means of image $\mathrm{U}(x, y)$ inside and outside the curve Γ, the term $\mathrm{Length}(\Gamma)$ denotes the length of Γ, the term $\mathrm{Area}(\Gamma)$ denotes the area inside Γ, and μ, ν, λ_1, λ_2 are positive parameters. Since then, many variants of the Chan-Vese model have been studied. For example, Yang *et al.* [27] proposed a high-order weighted variational model for image segmentation, the weights of which are automatically estimated based on edge information of the observed images. Wu *et al.* [28] introduced an effective regularization term, which combines an adaptive weighted matrix to enhance the diffusion along the tangent direction of the edge. Indeed, the AC model minimizes not only the distance between the solution and the input image, but also the length and area of the interfaces and foreground region, which has been used as the loss function for deep neural networks to discover better boundaries. Hu *et al.* [29] used the active contour model to help the deep network to learn information about the salient object. Marcos *et al.* [30] presented deep structured active contours to integrate priors and constraints, e.g., continuous boundaries, smooth edges and sharp corners, into the segmentation process. Chen *et al.* [31] proposed a loss function inspired by the general idea of the active contour model building in region and length terms for bop-medical image segmentation. Kim *et al.* [32] introduced a novel loss function to utilize spatial correlation in ground truth based on the level set formulation. Hatamizadeh *et al.* [33] introduced a deep active lesion segmentation model by making use of the precise boundary delineation abilities of the active contour model. Zhang *et al.* [34] integrated the convexified CV model [35] into the CNN structure to generate a more accurate segmentation of contours. Kim and Ye [36] proposed a new loss function based on the active contour model (2) for deep networks in semi-supervised and unsupervised manners. Ma, He and Yang [37] proposed a level set function regression network by minimizing the geodesic active contour energy in an end-to-end manner. However, the aforementioned losses may lose their effect for small organ segmentation, because less boundary information is available. In addition, we summarize the loss functions used for medical segmentation in Table 1, which can be divided into region loss, boundary loss, and their combination.

Aiming to aggregate complementary information from the image, especially pixel-level information, we propose both the multi-level structural loss function and a multi-level structural network to encode the multi-scale contextual features from different perspectives for realizing better small organ segmentation. To be specific, our loss function can measure the similarities between the prediction and ground truth from low-level pixel-wise classification to mid-level edge localization and to high-level region segmentation. In what follows, we develop a novel multi-level structural network, which incorporates three branches used to learn different features and uses a saliency guidance module to leverage the multi-scale information for small organ segmentation. We adopt the coarse-to-fine segmentation framework for automatic segmentation. Both coarse and fine models employ ResNet18 as the backbone and apply atrous spatial pyramid pooling (ASPP) module to facilitate multi-scale feature extraction and fusion. Our segmentation model is evaluated on two pancreas segmentation datasets, i.e., the NIH dataset [56] and the ISICDM dataset[1], and the spleen subset of the Medical Segmentation Decathlon (MSD) dataset[2]. By comparing with several state-of-the-art 2D and 3D learning-based segmentation approaches, our model is shown with better accuracy and higher efficiency.

The rest of the paper is organized as follows. Section 2 describes the details of our approach including the problem formulation and network architecture. We develop a multi-heads CNN to better fuse the multi-level structural features in Section 3. Section 4 is dedicated to providing the implementation details of our model. Evaluations and experimental results are presented in Section 5. Finally, we conclude the paper and discuss possible future works in Section 6.

2. Our segmentation framework

2.1. Our minimization problem

Considering that the image function can be measured on the region, contour and pixel-level, we propose the following multi-level structural loss to consist of complementary information for small organ or unbalanced segmentation

$$\min_{\theta} \; E_R(\mathrm{P}_\theta, \mathrm{V}) + E_B(\mathrm{P}_\theta, \mathrm{V}) + E_P(\mathrm{P}_\theta, \mathrm{V}), \tag{3}$$

where the terms $E_R(\mathrm{P}_\theta, \mathrm{V})$, $E_B(\mathrm{P}_\theta, \mathrm{V})$, and $E_P(\mathrm{P}_\theta, \mathrm{V})$ measure the differences between the prediction P and the ground truth V in the region, boundary, and pixel-wise level, respectively. As shown in Fig. 1, the first term can describe the overall appearance of the foreground, the other two terms can identify differences on boundaries. Unlike most existing loss functions focusing on the regional and edge information, our novel pixel-wise term penalizes on the

[1] http://www.imagecomputing.org/2018/challenge_CN.html
[2] http://medicaldecathlon.com/

Table 1
The loss functions used for medical image segmentation.

Category	Loss function	Illustration
Region loss	DSC [38]	Training the overlap region
	IoU [39]	Training the intersection-over-union
	Tversky loss[40]	Introducing a weight to balance the true negative and false positive cases based on DSC loss
	Generalized DSC [41]	Training the overlap regions of both foreground and background.
	Focal Tversky loss [42]	Combining the idea of focal loss and Tversky loss.
	Asymmetric similarity loss [43]	Introducing a weighting paramerter based on the Tversky loss
	Penalty loss [44]	Generalized Dice coefficient with a term for penalizing false negative and false positive
	weighted CE [8]	Cross entropy with class-balancing weight
	TopK loss [45]	Concentrating on the hard samples by dropping out the easy samples
	Focal loss [46]	Using a modulating factor to control the importance of easy samples based on cross entropy
	DPCE loss [47]	Using the distance map as the weights for cross entropy
	CE+DSC [48]	Combining cross entropy and DSC loss
	Focal loss+DSC [49]	Combining focal loss and DSC loss
Boundary loss	HD loss [50]	Weighted integrals over the interface between the regions
	Boundary loss [51]	Signed weighted integrals over the regions
	SDF regression[52]	Regressing the signed distance function
	Elastic interaction-based loss[53]	The elastic interaction energy between the boundary of the regions
	Contour DSC[54]	Measuring the distance between the surfaces by contour DSC
	Geometrically constrained Loss[55]	A geometrically constrained objective function using prior contour knowledge
Region loss & Boundary loss	Level set loss [17]	Using the nested level set to represent the region and edge locations
	DSC + SDF regression [37]	Learning the signed distance function and penalizing the overlap region
	CE + CE [15]	Using cross entropy to learn the region and boundary predictions

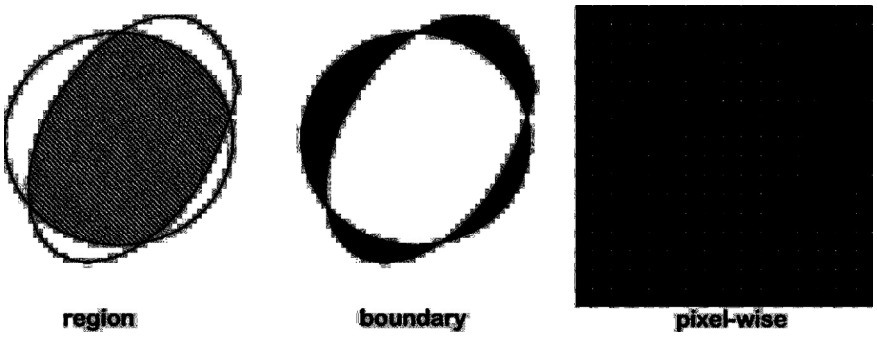


Fig. 1. Illustration of the multi-level structural information from global to local scopes, where the affinity between the prediction (red boundary) and the ground truth (blue boundary) can be depicted by region, boundary and pixel-wise measurements. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

uncertain points to capture the local features ignored by the region and boundary terms. By leveraging the three-level features, our method can ideally identify the structural information during the learning process and produce more desirable boundaries.

2.1.1. Region loss

The region term is widely used in segmentation tasks, which measures the overlap between the region predicted by the networks and the corresponding ground truth; see the blue region in Fig. 1 (a). There are various choices of the regional term, where we focus on binary segmentation for easy illustration. Suppose Ω is a bounded open subset of $\mathbb{R}^n$. Both the binary cross entropy (BCE)

$$\min_{\theta} -\sum_{\Omega} \mathrm{V} \odot \log(\mathrm{P}_{\theta}) - \sum_{\Omega} (1-\mathrm{V}) \odot \log(1-\mathrm{P}_{\theta}), \tag{4}$$

and Dice loss [38]

$$\min_{\theta} \; 1 - 2\frac{\sum_{\Omega} \mathrm{V} \odot \mathrm{P}_{\theta}}{\sum_{\Omega} \mathrm{V} + \mathrm{P}_{\theta}}, \tag{5}$$

are commonly used to train the CNN models for medical image segmentation tasks, where $\odot$ denotes the pixel-wise multiplication. The cross entropy aims to predict every pixel to find the target region, while the Dice loss cares about the overlap region. Many variants of BCE and Dice loss are studied for medical segmentation. Ronneberge *et al.* [8] introduced a class-balancing weight for cross entropy, which used the weight to balance the number of positive samples and negative samples. Caliva *et al.* [47] used the distance map as the weights for cross entropy, which focused on hard-to-segment boundary regions. The TopK loss [45] also concentrates on the hard samples by dropping out the easy samples. Lin *et al.* [46] proposed the focal loss for object detection first and then used in image segmentation, which introduced a modulating factor to control the importance of easy samples. The IoU loss [39] is designed similarly to Dice loss, where the intersection-over-union was minimized in deep neural networks. Sudre *et al.* [41] proposed the generalized Dice loss by minimizing the overlap regions of both foreground and background. Yang, Kweon and Kim [44] introduced a novel loss function by adding a term for penalizing false negative and false positive to generalized Dice coefficient. Salehi [40] presented the Tversky loss by introducing a weight to balance the true negative and false positive cases. Hashemi *et al.* [43] proposed the Asymmetric similarity loss based on the Tversky loss to achieve a better tradeoff between precision and recall. Abraham *et al.* [42] proposed a focal Tversky loss by combining the idea of focal loss and Tversky loss. The compound losses have also been implemented for segmentation such as the combination of Dice and cross entropy loss for nnU-Net [48], the combination of Dice and focal loss for anatomyNet [49], etc.

2.1.2. Boundary loss

For small organ segmentation and unbalanced segmentation, the region losses, e.g., Dice and CE, treat all the samples and

classes on an equal footing, which results in unstable training and predicted boundaries biased towards the majority classes. Kervadec *et al.*[51] proposed a boundary loss to use integrals over the interface between the regions instead of over regions, which is formulated as a distance metric on the space of contours (see Fig. 1 (b) for a visual illustration)

$$\min_{\theta} \sum_{\Omega} \mathrm{P}_{\theta} \odot \Phi, \tag{6}$$

where Φ is the signed distance function defined on the ground truth

$$\Phi(x) = \begin{cases} -\inf\limits_{y \in \partial \mathcal{D}} \|x-y\|_2, & \text{if } x \in \mathcal{D}; \\ 0, & \text{if } x \in \partial \mathcal{D}; \\ \inf\limits_{y \in \partial \mathcal{D}} \|x-y\|_2, & \text{if } x \in \Omega \backslash \mathcal{D}; \end{cases} \tag{7}$$

with $\mathcal{D}$ being the object region and $\partial \mathcal{D}$ being the interface. To reduce the Hausdorff distance between the predicted boundary and the target boundary, Karimi and Septimiu [50] used the following differentiable Hausdorff distance to train CNNs

$$\min_{\theta} \sum_{\Omega} (\mathrm{P}_{\theta} - \mathrm{V})^2 \odot (\mathrm{V}_{DTM}^2 + \mathrm{P}_{\theta,DTM}^2), \tag{8}$$

where V_{DTM} and P_{DTM} denote the distance transform maps of the ground truth V and prediction P with the form

$$\mathrm{V}_{DTM}(x) = \begin{cases} -\inf\limits_{y \in \partial \mathcal{D}} \|x-y\|_2, & \text{if } x \in \mathcal{D}; \\ 0, & \text{otherwise.} \end{cases}$$

Xue *et al.* [52] proposed to use CNNs to regress the signed distance function of ground truth. Moltz *et al.* [54] developed the contour Dice coefficient to quantify how much a surface of the predicted segmentation to the reference segmentation. Lan, Xiang and Zhang [53] proposed an elastic interaction-based loss function by minimizing the elastic energy of the curve for long thin structures. Azzopardi *et al.* [55] introduced geometrically constrained objective function which is constructed and tuned towards the segmentation of carotid structures using prior knowledge.

2.1.3. Pixel-wise loss

However, the value of the signed distance function (7) tends to zero as the pixels approach the boundaries, which results in inaccurate segmentation near organ boundaries. It is well-known boundaries depict important high-level details of the target. Therefore, we introduce the pixel-wise loss term to enhance the regularization effect on the pixel-level predictions. Bansal *et al.* [57] proposed the efficient PixelNet showing that a small number of pixels sampling from per image are sufficient for learning to achieve satisfactory segmentation performance. The merit of PixelNet is twofold, sampling only requires on-demand computation and offers the flexibility to allow the network to focus on rare samples. Kirillov *et al.* [58] presented the PointRend neural network module to perform point-based segmentation predictions at adaptively selected locations based on an iterative subdivision algorithm, where image segmentation is viewed as a rendering problem. The PointRend works as post-processing of CNN models, which can output high-resolution predictions over a finer grid to obtain sharp boundaries between objects. Apparently learning pixel-level information is important for both segmentation accuracy and model efficiency, especially when a coarse segmentation is already obtained.

2.2. Our coarse-to-fine model

We aim to employ 2D networks for segmenting 3D medical images such as the pancreas and spleen, from CT abdominal scans. For such a small organ segmentation problem, the coarse-to-fine approach is a good choice, which uses the prediction of the coarse stage to shrink the input for the fine stage [59,60].

We denote a scanned 3D image as $\mathbf{U}$ with the size of $W \times H \times L$, where W, H, and L are the number of slices along with the coronal, sagittal, and axial view, respectively. Then we slice each volume into 2D slices along each axis, which are denoted as $\mathrm{U}^{C,w}$ $(w = 1, \ldots, W)$, $\mathrm{U}^{S,h}$ $(h = 1, 2, \ldots, H)$, and $\mathrm{U}^{A,l}$ $(l = 1, 2, \ldots, L)$, respectively. Now, we take the axial view as an example to illustrate our coarse-to-fine model. The coarse model predicts a coarse segmentation from the input data, which can be formulated as

$$\mathrm{P}_{\theta_C}^{A,l} = \mathcal{F}_C(\mathrm{U}^{A,l}; \theta_C).$$

The merit of the coarse segmentation is that we can not only estimate a bounding box based on the segmentation but also generate a good initialization for the fine model. Firstly, we obtain a bounding box based on the binary segmentation $\widetilde{\mathrm{P}}_C^{A,l} = \mathbb{I}(\mathrm{P}_{\theta_C}^{A,l} \geq 0.5)$, which is a minimal 2D bounding box containing all nonzero pixels of $\widetilde{\mathrm{P}}_C^{A,l}$ and a K-pixel wide margin. Then, we define a crop function $\mathcal{C}[\cdot; \widetilde{\mathrm{P}}_C^{A,l}]$ to crop the bounding-box region from a given image with the same size as $\mathrm{P}_{\theta_C}^{A,l}$. On the other hand, followed [60], we introduce the saliency transformation module to generate an image with attention as the initialization to the fine model

$$\mathrm{I}_{\theta_S}^{A,l} = \mathcal{F}_S(\mathrm{P}_{\theta_C}^{A,l}; \theta_S),$$

where $\mathcal{F}_S(\cdot; \theta_S)$ is the transformation function parameterized by θ_S. We further use the fine model to estimate the final segmentation

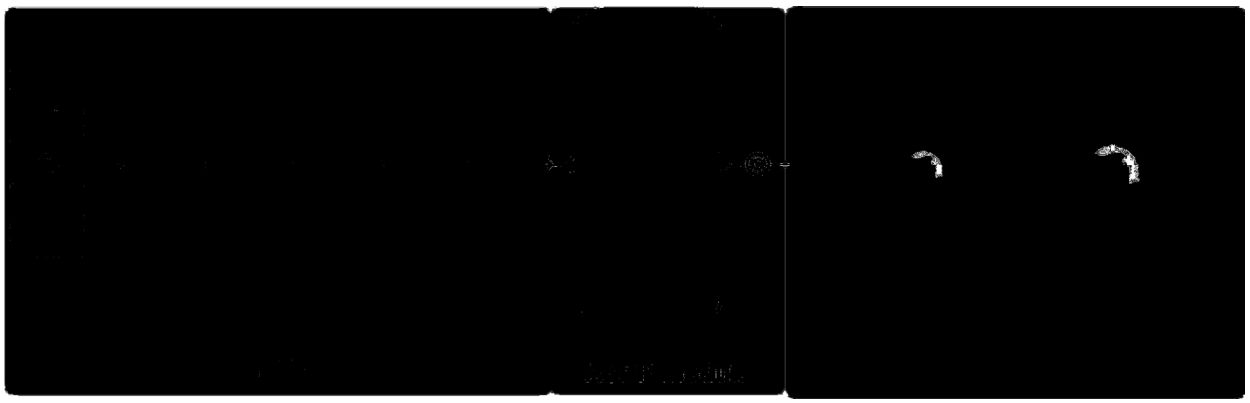


Fig. 2. Illustration of the fine segmentation model using the multi-level structural loss, where 1×1 stands for the convolution with kernel 1. Although a 3-slice unit is used in the network, we only display one slice for ease of understanding.

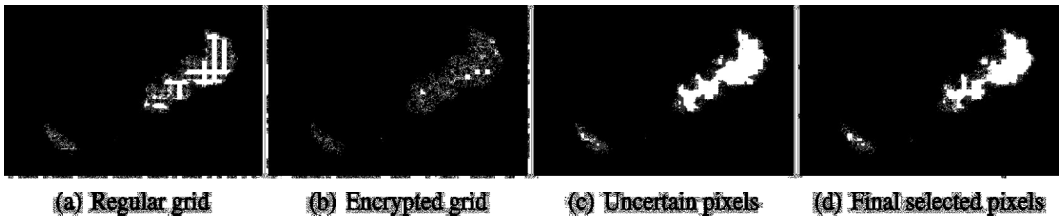


Fig. 3. A typical visual example of the selection procedure of the uncertain pixel set $\mathcal{S}$ on the final logits layer.

such as

$$\mathrm{P}_{\theta}^{A,l} = \mathcal{F}\big(\mathcal{C}[\mathrm{I}_{\theta_S}^{A,l} \odot \mathrm{U}^{A,l}; \widetilde{\mathrm{P}}_{C}^{A,l}], \mathcal{C}[\mathrm{U}^{A,l}; \widetilde{\mathrm{P}}_{C}^{A,l}]; \theta\big), \tag{9}$$

where $\mathcal{F}(\cdot;\theta)$ is our segmentation function parameterized by θ. Note that we concatenate the original image $\mathrm{U}^{A,l}$ into the input of the fine model to enhance the shallow features such as edges, lines and corners. It is not difficult to find out that the coarse and fine models are jointly together through the saliency transformation. Our final segmentation is the binary image obtained by $\widetilde{\mathrm{P}}^{A,l} = \mathbb{I}(\mathrm{P}_{\theta}^{A,l} \geq 0.5)$.

2.3. Network architecture

For the task of small organ segmentation, we build up both coarse and fine models by carrying forward the cascade blocks and the ASPP module in DeepLab-v3 to go deeper with atrous convolution and pyramid pooling. For illustration purposes, we go through the network architecture of our fine model, which is displayed in Fig. 2. We employ ResNet-18 as our backbone network, which contains four basic blocks. The input of the fine model contains both saliency map and original image to provide a good initialization and more image features. Before entering block1, we remove the downsampling operation in the first convolution layer to keep the information inside the input. We use the atrous convolution with rate=2 and rate=4 to replace the consecutive striding in block3 and block4, respectively. The motivations behind these operations are to preserve more local information and enlarge the receptive field, which are important for the small target. The ASPP module can effectively capture multi-scale contextual information by integrating features obtained with different atrous rates. By further passing the features through two 1×1 convolutional layers, we obtain the final logits L_{θ}, which is used to generate the prediction by upsampling using the bilinear interpolation. For the coarse network, the only difference is we keep the striding in the first convolution layer to reduce computational cost. Lastly, for the saliency transformation, we simply apply two size-preserved convolutions with filter size 5×5 and 3×3, respectively.

2.4. Loss function

We use the Dice loss and multi-level structural loss (3) for the coarse and fine model, respectively. As illustrated in Fig. 2, the region and boundary term in the multi-level loss is chosen as Dice and boundary loss in [51], which are evaluated to be a powerful combination for small organ segmentation [61]. Because the spatial redundancy limits the information learned by convolutional networks for general pixel-level prediction problems, the final prediction P_{θ} of the networks tends to over smooth the organ region and under sample organ boundary. Thus, we define the pixel-wise loss term on the final logits layer L_{θ} (see Fig. 2 for illustration), which is $4\times$ coarser than the image grid. In particular, we define an uncertain pixel set $\mathcal{S}$ on the logits layer by involving pixels with the predicted probabilities closest to 0.5, which locate near the boundaries and are most difficult to segment. By minimizing the binary cross entropy on the uncertain set $\mathcal{S}$, we obtain the following pixel-wise contextual loss

$$\begin{aligned} E_P(\mathrm{L}_{\theta}, \mathrm{V}) = &-\sum_{\Omega} \mathrm{V} \odot \log(\mathrm{L}_{\theta}) \odot \mathcal{I}_{\mathcal{S}}(\mathrm{L}_{\theta}) \\ &-\sum_{\Omega} (1-\mathrm{V}) \odot \log(1-\mathrm{L}_{\theta}) \odot \mathcal{I}_{\mathcal{S}}(\mathrm{L}_{\theta}), \end{aligned} \tag{10}$$

with $\mathcal{I}_{\mathcal{S}}(\mathrm{L}_{\theta})$ being the indicator function defined as

$$\mathcal{I}_{\mathcal{S}}(\mathrm{L}_{\theta}) = \begin{cases} 1, & \text{for } x \in \mathcal{S}; \\ 0, & \text{for } x \notin \mathcal{S}. \end{cases}$$

The selection of the uncertain pixel set is realized in a three-step strategy similar to [58]. To obtain a sharp and accurate boundary, we first encrypt the pixels of the final logits layer L_{θ} by randomly sampling kN pixels with $k = 3$ followed a uniform distribution to generate enough pixels with uncertainty. Next, we estimate the probabilities on the kN pixels by interpolating the original prediction. Finally, we choose mN pixels with $m = 0.75$ of the highest uncertainty and $(1-m)N$ random pixels to increase the generalization of the data. Thus, our uncertain points set contains N points in total. From the example in Fig. 3, we can find out that the uncertain pixels are also located near the boundaries. Although the boundary loss can improve accuracy by minimizing the differences between the prediction and ground-truth on the boundary, it treats the boundary as a whole by missing the pixel-level consideration. We introduce the pixel-wise loss on a coarse segmentation and gradually raise the regularization to allow the networks to learn more pixel-level information. By allocating the pixels on a fine grid, our pixel-wise loss term can not only promote the segmentation accuracy on uncertain pixels, but also provide smoother boundaries, which works complementary to the boundary loss. Moreover, our pixel-wise loss term is flexible to combine with other region and boundary loss terms and implement to any existing deep network architecture.

Per the previous discussion, we jointly minimize the following energy functional in the training stage to realize efficient multi-level feature fusion

$$\begin{aligned} \mathcal{L}_{MLL}(\theta_C, \theta) = &\omega_1 E_R\big(\mathrm{P}_{\theta_C}, \mathrm{V}\big) + \omega_2 [E_R(\mathrm{P}_{\theta}, \mathrm{V}) \\ &+ \alpha E_B(\mathrm{P}_{\theta}, \mathrm{V}) + \beta E_P(\mathrm{L}_{\theta}, \mathrm{V})], \end{aligned} \tag{11}$$

where $\omega_1, \omega_2, \alpha, \beta \in \mathbb{R}$ are parameters to balance the contributions of different terms.

3. Multi-level structural network

Inspired by the success of the multi-task models [19,37], we develop a novel network working together with our multi-level structural loss to learn and fuse image features. As shown in Fig. 4, our multi-level structural network uses multiple branches to learn features from region-level to edge-level and pixel-wise level, respectively, and adopts a transformer structure to capture long-range information from the carefully chosen points. All three branches are processed in parallel and fused together by a novel saliency guidance module, which can leverage the multi-level features and out-

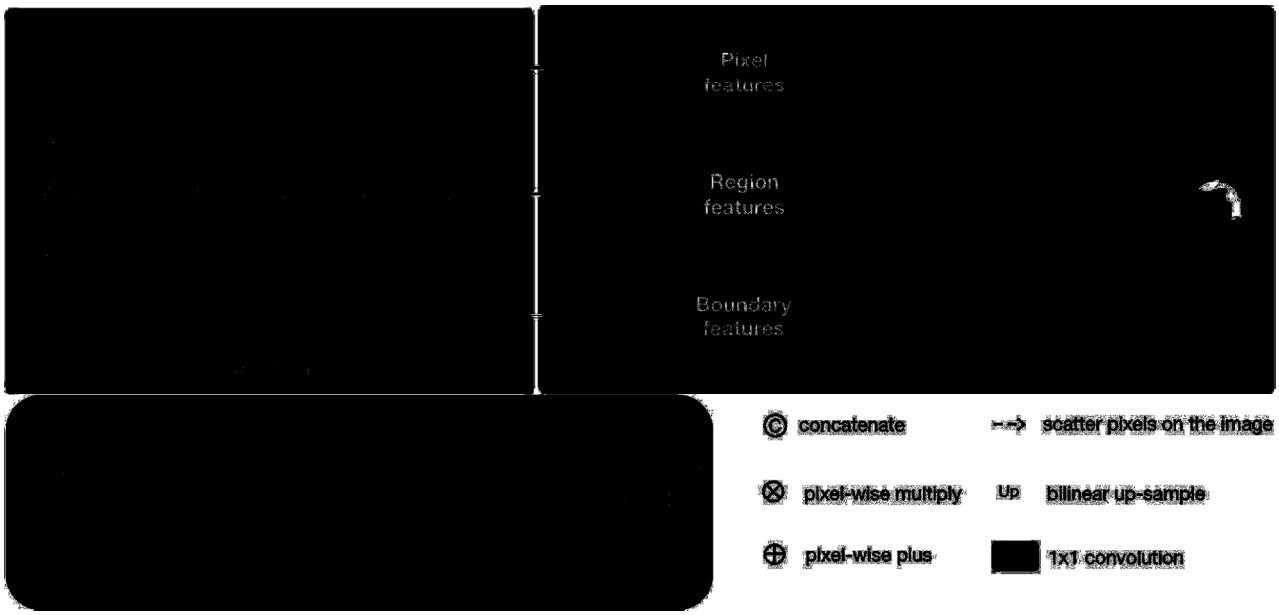


Fig. 4. Illustration of our multi-level structural network. Although a 3-slice unit is used in the network, we only display one slice for ease of understanding.

put the final prediction. In the following, we explain the branches one by one in details.

3.1. The region branch

The region branch adopts ASPP module as mentioned above, which can capture image context at multiple scales and outputs the binary segmentation prediction $\mathrm{P}_{\theta}^{region}$. During the training, we use the Dice loss to penalize the regional prediction.

3.2. The boundary branch

We implement another ASPP module as the boundary branch in favor of its ability in resampling features at different scales. Unlike most boundary prediction models, which outputs a binary map with 1 indicating the edges and 0 otherwise, ours learns a signed distance map defined by (7). As in the previous step, we define the boundary loss function as the combination of boundary loss and an ℓ_2 norm of the signed distance map, which gives

$$E_B^S(\mathrm{P}_{\theta}^{boundary}, \mathrm{V}) = E_B(\sigma(\mathrm{P}_{\theta}^{boundary}), \mathrm{V}) + \frac{1}{n}\sum \|\mathrm{P}_{\theta}^{boundary} - (-\Phi)\|^2, \tag{12}$$

where the $\sigma(\cdot)$ represents the sigmoid function used to transform the signed distance map into the segmentation prediction. Note that the boundary prediction $\mathrm{P}_{\emptyset}^{boundary}$ is positive inside the target and negative outside the target. Thus, there is a '-' (minus) sign before Φ.

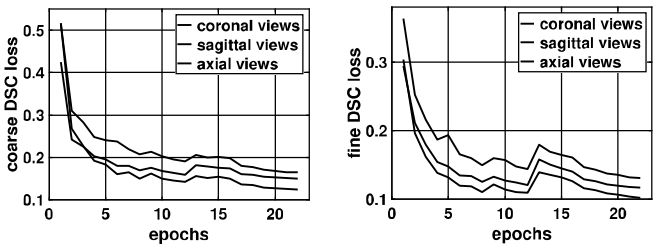
3.3. The pixel branch

For the pixel branch, we use a non-local transformer module to learn one-dimensional pixel-wise features, which not only leverages local feature and global context, but also emphasizes the characteristic by the self-attention mechanism. The pixel branch directly extracts features from the ResBlock1 of the backbone to enhance the low-level features. We estimate the position of the uncertain points based on the region prediction, where the uncertain points selection method is the same as previous except for the encryption. In particular, we choose the uncertain points based on the regular points rather than encrypted points to facilitate the integration with the saliency guidance module. PointRend [58] used the shared MLPs to predict pixel-wise segmentation, which can only extract local features. Our pixel branch adopts the transformer architecture to catch the non-local features. As we can see from the green box in Fig. 4, our pixel branch contains a transformer and a shared MLP to leverage the local and nonlocal features for final prediction. The transformer consists of a multi-headed self-attention block similar to [62] followed by a point-wise MLP layer (1×1 convolution). Then the ReLu activation function is applied after the point-wise MLP layers. We also incorporate the residual connections. As discussed, the binary cross entropy is used as the loss function for the pixel branch to penalize the predictions of the selected points $\mathrm{L}_{\emptyset}^{pixel}$.

3.4. The multi-level saliency guidance module

Finally, we employ a multi-level saliency guidance module to fuse different features together for final predictions. The features before the prediction outputted by all branches are all of 256 channels. We then upsample the features of the region and boundary branch to the original size of the input images. The features of the uncertain points are one-dimensional data of 256 channels. We first generate the pixel features with the same size as the region and boundary level features by filling with zero values, and then replace the position of the uncertain points by learned features, which means only the selected uncertain positions are of nonzero values. In our multi-level saliency guidance module, both boundary features and pixel features are element-wise summarized together and two depth-wise 1×1 convolutions are adopted to extract the mutual relation between the boundary and pixel features to generate two weight maps, which are used to multiply with the boundary and pixel features to obtain the saliency maps, respectively. Afterward, the region features are multiplied with the two saliency maps individually, which are then concatenated together. By two 1×1 convolutions, we obtain the final prediction $\mathrm{P}_{\theta}^{fusion}$. We use the regional Dice loss to train the multi-level saliency guidance module.

We use the aforementioned multi-level structural loss to learn plentiful features, where the boundary loss can alleviate the im-



(a) Coarse model (b) Fine model

Fig. 5. The line chart of the DSC loss during training. When adding our pixel-wise loss, the DSC loss increase first, but decrease soon in the rest epochs.

balance issue on edge prediction and the pixel loss supervising on selected points can remedy local information. To be specific, the loss function used to train our multi-level structural network can be described as follows:

$$\mathcal{L}_{MLN}(\theta_C, \theta) = \omega_1 E_R(\mathrm{P}_{\theta_C}, \mathrm{V}) + \omega_2[E_R(\mathrm{P}_{\theta}^{region}, \mathrm{V}) + \alpha E_B^S(\mathrm{P}_{\theta}^{boundary}, \mathrm{V}) + \beta E_P(\mathrm{L}_{\theta}^{pixel}, \mathrm{V}) + \gamma E_R(\mathrm{P}_{\theta}^{fusion}, \mathrm{V})], \quad (13)$$

where $\omega_1, \omega_2, \alpha, \beta, \gamma \in \mathbb{R}$ are parameters to balance the contributions of different branches.

4. Implementation details

All our experiments are implemented with Pytorch on NVIDIA Titan RTX.

4.1. Network training and testing

4.1.1. The training phase

We jointly train the coarse and fine models in the training stage. Followed [63], we use the three-step optimization strategy, which can guarantee the convergence of the coarse-to-fine approach. Firstly, we use the ground truth to generate the bounding box and optimize the coarse and fine model separately. Secondly, we still use the ground truth bounding box, but jointly optimize the coarse and fine models through the saliency transformation module. Finally, we directly optimize the coarse-to-fine network without using the ground truth bounding box. The parameters ω_1 and ω_2 in our loss function (13) are selected as $\omega_1 = 0.3$, $\omega_2 = 2/3$ to balance the contributes of the coarse and fine models. For the multi-level structural loss model, the parameter α is initialized as $\alpha = 0$ and adjusted dynamically such that α is gradually increased by 0.2 each epoch until reaching $\alpha = 1$. Our pixel-wise loss is considered when the predictions are with certain accuracies to guarantee the convergence of the entire network. Thus, the value of β is set to $\beta = 0$ during the first 12 epochs and then increased to $\beta = 1$ for the remaining epochs. The curves of DSC loss for both coarse and fine models along three axes views are plotted in Fig. 5. As can be seen, the value of DSC converges as the number of epochs increases. There are two obvious vibrations for both coarse and fine models. At the 8th epoch, we start to use the coarse segmentation to draw the bounding box instead of the ground truth bounding box, which results in the rise of the values of DSC for both coarse and fine models. Another one is the 12th epoch, where we introduce the pixel-wise loss into our loss function. However, after several epochs, the value of DSC decreases to much lower values, which demonstrates the pixel-wise loss can help the convergence of the proposed model. In the training, we adopt an early-stopping scheme by stopping training when there is no improvement in the performance on the validation set.

On the other hand, the parameter setting for our multi-level structural network is given in a similar way. The parameter α is initialized as $\alpha = 1$ since the region branch and the boundary branch are trained separately. The value of β and γ is set as $\beta = \gamma = 0$ for the first 12 epochs and then increased to $\beta = \gamma = 1$ for remaining epochs, which can guarantee the convergence of the pixel branch and the final prediction.

4.1.2. The testing phase

Different from the training process, our testing stage is implemented in an iterative procedure to progressively refine the output. The coarse segmentation, denoted as $\mathbf{P}_0$, is used to estimate the bounding box and saliency map, both of which are the input for the fine model. Then the fine model iteratively updates the bounding box and saliency map until the convergence. We use both the maximum iteration and the relative DSC to track the convergence, where the RDSC is defined as

$$\mathrm{RDSC}(\widetilde{\mathbf{P}}_t, \widetilde{\mathbf{P}}_{t+1}) = \frac{2|\widetilde{\mathbf{P}}_t \cap \widetilde{\mathbf{P}}_{t+1}|}{|\widetilde{\mathbf{P}}_t| + |\widetilde{\mathbf{P}}_{t+1}|} \leq tol, \quad (14)$$

with $tol = 0.99$ in our experiments, and the maximum iteration number is $T_{\max} = 10$. We sketch the algorithm in the testing as Algorithm 1 .

Algorithm 1: The testing stage.

Input: CT image $\mathbf{U}$, model $\mathcal{F}_C(\cdot)$, $\mathcal{F}_S(\cdot)$, $\mathcal{F}(\cdot)$, $t = 0$, $\mathrm{RDSC}(\widetilde{\mathbf{P}}_0, \widetilde{\mathbf{P}}_1) = 1$;
Output: Segmentation $\mathbf{P}$;
/* Coarse model */
for $l = 1, \ldots, L$ **do**
  $\mathrm{P}_0^{A,l} \leftarrow \mathcal{F}_C(\mathrm{U}_C^{A,l}; \theta_C)$;
  $\mathrm{I}_0^{A,l} \leftarrow \mathcal{F}_S(\mathrm{P}_0^{A,l}; \theta_S)$;
  $\widetilde{\mathrm{P}}_0^{A,l} \leftarrow \mathbb{I}[\mathrm{P}_0^{A,l} \geqslant 0.5]$;
end
/* Fine model */
while $RDSC(\widetilde{\mathbf{P}}_t, \widetilde{\mathbf{P}}_{t+1}) < 0.99$ *or* $t < T_{\max}$ **do**
  for $l = 1, \ldots, L$ **do**
    $\mathrm{P}_{t+1}^{A,l} \leftarrow \mathcal{F}\big(\mathcal{C}[\mathrm{I}_t^{A,l} \odot \mathrm{U}^{A,l}; \widetilde{\mathrm{P}}_t^{A,l}], \mathcal{C}[\mathrm{U}^{A,l}; \widetilde{\mathrm{P}}_t^{A,l}]; \theta\big)$;
    $\mathrm{I}_{t+1}^{A,l} \leftarrow \mathcal{F}_S(\mathrm{P}_{t+1}^{A,l}; \theta_S)$;
    $\widetilde{\mathrm{P}}_{t+1}^{A,l} \leftarrow \mathbb{I}[\mathrm{P}_{t+1}^{A,l} \geqslant 0.5]$;
  end
  $\mathrm{RDSC}(\widetilde{\mathbf{P}}_t, \widetilde{\mathbf{P}}_{t+1}) \leftarrow \frac{2|\widetilde{\mathbf{P}}_t \cap \widetilde{\mathbf{P}}_{t+1}|}{|\widetilde{\mathbf{P}}_t| + |\widetilde{\mathbf{P}}_{t+1}|}$;
  $t \leftarrow t + 1$;
end
return $\mathbf{P} \leftarrow \widetilde{\mathbf{P}}_{t+1}$.

Besides, for dealing with 3D medical image segmentation, we usually train three models along with each view and fuse the segmentation results by majority voting first. More details can be found in [59,60].

4.2. Architecture setting

We first discuss how to modify the architecture of ResNet backbone to overcome its disadvantage in dealing with small organ segmentation. The two down-sampling operations before the first block can decrease the computational cost by sacrificing sharp details. Thanks to the coarse-to-fine structure of our model, the inputs of the fine model is already with the size less than 1/2 of their original sizes. Thus, it is better to remove the down-sampling operations for catching more detail information from the image. Let *out_stride* denote the ratio of the input image spatial resolution to output resolution. By removing either the max-pooling layer or the striding in the first convolution or their combination, we can

Table 2
Segmentation accuracy comparison with respect to different values of *out_stride*, where N/A denotes without the max-pooling layer.

	conv1	max-pool	block1	block2	block3	block4	DSC	Time per epoch	Parameters	FLOPs
atrous	1	1	1	1	2	4	84.51%	492s	9.2×10^7	1.34×10^{11}
out_stride	2	4	4	8	8	8				
atrous	1	N/A	1	1	2	4	85.49%	496s	9.2×10^7	1.49×10^{11}
out_stride	2	N/A	2	4	4	4				
atrous	1	1	1	1	2	4	85.83%	497s	9.2×10^7	1.49×10^{11}
out_stride	1	2	2	4	4	4				
atrous	1	N/A	1	1	2	4	85.69%	607s	9.2×10^7	2.07×10^{11}
out_stride	1	N/A	1	2	2	2				

obtain different *out_stride*. We compare the segmentation accuracy, training time per epoch and FLOPs (multiply-adds) with respect to the different *out_stride* in Table 2. As can be seen, the optimal choice is *out_stride* $= 4$ obtained by removing the down-sampling in the first convolution layer and keeping the max-pooling layer, for which both FLOPs and training time are close to the baseline model.

4.3. Datasets and settings

4.3.1. NIH pancreas dataset

The NIH pancreas dataset contains 82 contrast-enhanced abdominal CT volumes, the resolutions of which are $W \times H \times L$ volume with $W = H = 512$ and $L \in [181, 466]$. Similar to [60], we divide the dataset into 4 fixed folds, each of which contains almost the same number of samples. We use cross-validation to evaluate the segmentation performance, i.e., train the network on 3 out of 4 subsets and test it using the remaining one. In the data pre-processing step, we clip the image intensities within [-100, 240]. For the NIH dataset, we independently train three 2D models along each axis, i.e, the coronal, sagittal, and axial view, and then fuse the predictions in each iteration by the majority voting. The Resnet-18 pre-trained on Imagenet dataset is used to initialize the backbone and Adam is adopted as the optimizer. We train about 20 epochs with the learning rate being 1e-5 for the first 8 epochs and decreased by the rate of 1/2 in every 2 epochs. When introducing the pixel-wise loss at the 12th epoch, we reassign the learning rate to 1e-5 and also decrease it with the rate of 1/2 every two epochs. The batch size is set to 1 with 3 slices concatenated together. During the testing, although the segmentation accuracy keeps increasing in the iterative process, we stop it to balance the trade-off between the computational efficiency and segmentation accuracy, where the relative DSC tolerance is set to $tol = 0.99$ and the maximal iteration number is fixed as $T_{\max} = 10$. In both training and testing, we crop the pancreas region with a margin of $K = 20$.

We use a similar training method for our multi-level structural network. During the first 12th epochs, we omit both pixel branch and saliency guidance module. When the backbone of the fine model converges, we start to train the pixel branch and the saliency guidance module, which can ensure the uncertain points being properly chosen. The total number of epochs is set as 25 for the multi-level structural network. The results of nnU-Net [48] were re-implemented by ourselves, where the cascade 3D architecture was chosen for small organ segmentation. We trained five-fold for each cross-validation, and used the five networks obtained from the training set as an ensemble to estimate the results, whose pre-processing, post-processing and data augmentation all follow the suggestion in the original paper.

4.3.2. ISICDM pancreas segmentation challenge dataset

The ISICDM pancreas segmentation challenge contains 36 thin and 36 thick abdominal CT volumes. The scanning device is Definition AS, Siemens, under standard pancreas scanning protocol. The number of 2D slices ranges between [205,376] and [31,76] and the thickness is 1 mm and 3 mm for the thin and thick dataset, respectively. Following the cross-validation strategy, we split both datasets into 6 subsets, each subset of which contains 6 volumes. We use two models to process thin and thick datasets, both of which are trained on 5 out of 6 subsets and tested on the remaining subset. For the thin dataset, we also fuse the three 2D models along each axis as our final model. For the thick dataset, we use the axial view model because too few slices are contained in coronal and sagittal views resulting in bad predictions. The weights of both thin and thick models are pre-trained on the NIH dataset. We re-implemented both RSTN [60] and nnU-Net [48], where RSTN was trained using the same data pre-processing and pre-trained on the NIH dataset for better performance. For nnU-Net, we used the 3D cascade architecture and 3D full resolution architecture to process ISICDM thin and thick dataset, respectively. We also trained five nnU-Net models for each cross-validation with the same pre-processing, post-processing and data augmentation as the original paper.

4.3.3. Medical segmentation decathlon (MSD) dataset

The third one is the Medical Segmentation Decathlon (MSD) spleen dataset, which contains 41 CT volumes. The number of 2D slices ranges between [31,168]. Following the setting suggested in [64], we first clip the intensities of all images into [-125, 275] and randomly divide the dataset into two groups, one group containing 21 volumes used for training and the other one with 20 volumes used for testing. Similar to the experiment on the thick dataset of the ISICDM, we implement the one-dimensional model trained on the axial view for the prediction. Other settings are the same as the NIH dataset. For a fair comparison, we re-implemented the comparative models by ourselves. Following the settings of V-Net in [64], we normalized the volumes, and used $128 \times 128 \times 64$ patches in both training and testing. We also re-implemented the 3D cascade architecture of nnU-Net on the MSD spleen dataset with five network models obtained on the training set.

4.4. Evaluation

We use both Dice Similarity Coefficient (DSC) and Hausdorff distance (HD) to evaluate the performance of our model, which are defined as

$$\mathrm{DSC} = (\mathbf{P}, \mathbf{V}) = \frac{2|\mathbf{P} \cap \mathbf{V}|}{|\mathbf{P}| + |\mathbf{V}|},$$

and

$$d_H(\mathbf{P}, \mathbf{V}) = \max\left(\max_{x \in \partial\mathbf{P}} \min_{y \in \partial\mathbf{V}} \|x - y\|^2, \max_{y \in \partial\mathbf{V}} \min_{x \in \partial\mathbf{P}} \|x - y\|^2\right),$$

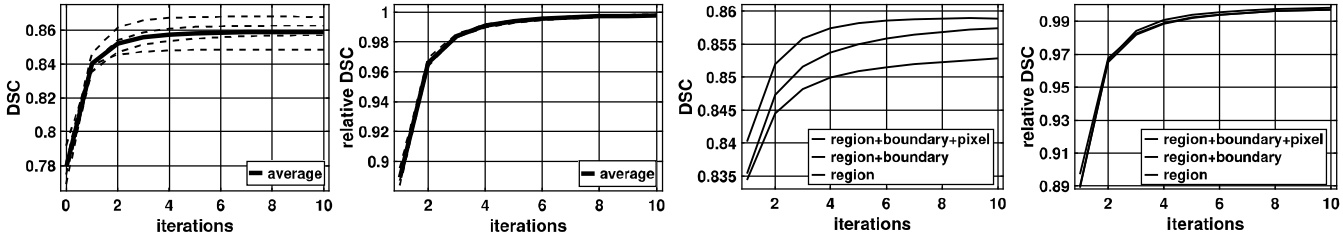
respectively. Theoretically, high DSC and low HD indicates better segmentation accuracy.

5. Experimental results

In this section, we evaluate both multi-level structural loss (MLL) model and multi-level structural network (MLN) model on

Table 3
Segmentation comparison between our model and the state-of-the-arts on NIH pancreas dataset.

Bounding box	Method	Year	DSC	Min	Max	Model type
with label	Liu *et al.*[65]	2018	86.70% ± 3.57%	73.61%	N/A	2-D
	Li *et al.*[66]	2020	87.57% ± 3.26%	73.68%	**93.40**%	2-D
	MLL model	–	**88.01% ± 2.50%**	79.80%	92.71%	2-D
	MLN model	–	87.87% ± 2.51%	**80.57%**	92.62%	2-D
w/o label	Zeng *et al.* [67]	2019	83.0% ± 5.85%	68.39%	90.31%	3-D
	Zhou *et al.* [68]	2018	84.59% ± 4.86%	69.62%	91.45%	3-D
	nnU-Net [48]	2021	84.98% ± 5.67%	60.66%	91.68%	3-D
	Xia *et al.* [69]	2018	84.63% ± 5.07%	61.58%	91.57%	2-D&3-D
	Chen *et al.* [70]	2019	85.22% ± 4.07%	**71.40%**	91.36%	2-D&3-D
	Zhou *et al.* [59]	2017	82.37% ± 5.68%	62.43%	90.85%	2-D
	RSTN [60]	2018	84.50% ± 4.97%	62.81%	91.02%	2-D
	Hu *et al.* [71]	2021	85.49% ± 4.77%	67.19%	91.64%	2-D
	MLL model	–	**85.83%** ± 4.37%	65.19%	91.71%	2-D
	MLN model	–	85.62% ± 4.56%	64.20%	**91.81%**	2-D



(a) DSC (b) RDSC (c) DSC (d) RDSC

Fig. 6. The first two line charts are the line chart of DSC and relative DSC. The dotted lines are the iteration results of 4 cross validations, and the red line is the average results. In the first 5 iterations, the DSC and relative DSC grow fast, and in the next 5 iterations, both the DSC and relative DSC continue growing, although the speed is slow down. The last two line charts are the ablation analysis of the multi-level structural loss on the NIH dataset. (a) the plots of DSCs w.r.t. 4 cross validations; (b) the plots of relative DSCs w.r.t. 4 cross validations; (c) the plots of DSCs w.r.t. different loss functions; (d) the plots of relative DSCs w.r.t. different loss functions. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

two pancreas segmentation datasets and a spleen dataset by comparing with several state-of-the-art segmentation approaches.

5.1. NIH pancreas dataset

5.1.1. Numerical analysis

A series of deep learning-based methods have been developed for NIH pancreas segmentation. Some works directly use ground-truth to generate a bounding box, while others use the multi-model methods to learn a bounding box. For a fair comparison, we train two kinds of models with or w/o label data for choosing the bounding box. The numerical results are summarized in Table 3, where our models surpass the state-of-the-art results for both situations. In particular, with the same cropping as done in [65], our method provides a significantly higher DSC value, which can demonstrate the advantages of the efficiency of the network structure and multi-level structural loss function. It demonstrates that our network architecture can catch more information than the encoder-decoder structure, which is useful for small organ segmentation such as the pancreas. On the other hand, both MLL and MLN models outperform the multi-model approaches, i.e., the combined 2D and 3D volumetric fusion models [69], and two recently published multi-model methods [48,71]. Most importantly, our MLL model surpasses the 3D multi-model method nnU-Net with an almost 1% higher DSC, which proves the effectiveness of the multi-level structural loss in small organ segmentation. Moreover, both our multi-level structural loss model and multi-level structural network gain a significant improvement ($\geq 1\%$) over the RSTN proposed by Yu *et al.* [60], which is also developed based on the coarse-to-fine and saliency transformation architecture. More specifically, we observe that our coarse model, without multi-level structural loss, gives an average segmentation accuracy of 77.96%, slightly lower than 78.23% in [60], which is built up with the FCN. However, our fine model provides much better accuracy, which achieves a DSC of 84.02% after the first iteration, much higher than 82.73% in [60]. It clearly shows that both multi-level structural loss and network can help the backbone model to gain better accuracy.

5.1.2. Convergence analysis

We further analyze the convergence of our multi-level structural loss model to verify its reliability. In particular, we track both absolute DSC and relative DSC in the testing stage. As shown in Fig. 6(a,b), both the absolute error and relative error of the four cross-validations converge as the number of iteration keeps increasing. Especially, we observe that both absolute and relative DSC grow fast in the first 5 iterations and almost remain the same in the last 5 iterations. Thus, it is reasonable to set $T_{\max} = 10$ in the testing stage.

On the other hand, we compare the relative DSC of our model and the two recurrent saliency transformation networks in Table 4. As can be observed, our model with multi-level structural loss converges fastest among all the compared methods, including both the two recurrent saliency transformation networks and our model with the reduced loss functions. As reported in [60], it requires 5.22 iteration on average to surpass the 0.99 relative DSC, while our model only needs 4.29 iterations. This means that our model can not only give predictions with higher DSC but also save certain computational costs in processing the data. It is worthy to mention that there are 2 out of 82 cases do not converge after 10 iterations while all cases converge using our multi-level structural loss. Furthermore, we list both DSC and HD versus iteration of our model in Table 5 and 6, respectively. We observe that the pixel-wise loss works well in improving the segmentation accuracy and the convergence of the proposed model.

Table 4
Convergence comparison between our model and the recurrent saliency transformation networks, which is measured by RDSC (%).

Method	1st iterate	2nd iterate	3rd iterate	5th iterate	10th iterate	Converged cases	Converged
RSTN [60]	0.9037	0.9637	0.9814	0.9908	0.9964	80/82	5.22 iters
Region loss	0.8970	0.9676	0.9826	0.9921	0.9970	80/82	4.85 iters
Region + boundary loss	0.8898	0.9652	0.9819	0.9918	0.9970	81/82	4.83 iters
MLL	0.8891	0.9662	0.9840	0.9939	0.9979	82/82	4.29 iters

Table 5
DSC (%) comparison of our model with respect to different level loss function versus iterations.

Method	1st iterate	2nd iterate	3rd iterate	5th iterate	10th iterate	Converged
Region loss	83.44%	84.45%	84.82%	85.09%	85.28%	85.20%
Region + boundary loss	83.54%	84.73%	85.16%	85.50%	85.74%	85.63%
MLL	84.02%	85.20%	85.58%	85.82%	85.89%	85.83%

Table 6
95% Hausdorff distance (mm) comparison of our model with respect to different level loss function versus iterations.

Method	1st iterate	2nd iterate	3rd iterate	5th iterate	10th iterate	Converged
Region loss	6.678	6.367	6.177	6.003	5.543	5.645
Region + boundary loss	6.405	6.022	5.798	5.538	5.305	5.366
MLL	5.751	5.419	5.240	5.082	5.045	5.069

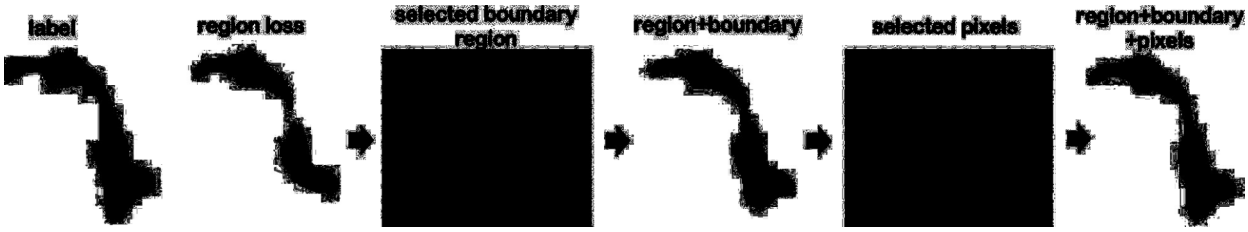


Fig. 7. A typical visual example of our multi-level structural loss. The selected boundary and pixels are displayed with the blue background. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

5.1.3. Ablation analysis

In what follows, we conduct a series of ablation studies to investigate the effectiveness of the region, boundary, and pixel-wise terms. As shown in Fig. 6 (c), each term in our multi-level structural loss contributes to the final segmentation performance. By introducing the boundary loss, the DSC raises by 0.43%. When introducing the pixel-wise loss, the DSC is further improved by 0.2%. Simultaneously, the Hausdorff distance used to measure the accuracy of boundaries also decreases by about 0.6 (mm) compared to the one obtained by the region loss. On the other hand, Fig. 6(d) confirms that learning multi-level features can help to improve the convergence of our coarse-to-fine model. Besides, we provide a visual illustration of the ablation experiment in Fig. 7, where the pancreas is of a long and narrow structure. As shown, by the region loss, the pancreas has been segmented into two separated subregions. By introducing the boundary loss, the isolated two subregions become connected, while the pixel-wise loss can further improve the accuracy of the boundary.

5.2. ISICDM pancreas dataset

We further discuss the segmentation performance on the ISICDM pancreas dataset. We trained two models for thin and thick data, respectively, Note that the thin model is pre-trained on the NIH dataset and the thick model is initialized with the thin model. The specific segmentation results are displayed in Table 7. We re-implemented the RSTN [60] and nnU-Net [48] for a fair comparison. As can be seen, our models not only provide better segmentation accuracy on both thin and thick datasets, but also save much computational time, especially compared to nnU-Net. When compared with RSTN, our model converges within 3.92 and 4.42 iterations, respectively, while the RSTN consumes 6.11 iterations on average to reach 0.99 relative DSC on the thin dataset. Consequently, much computational time is saved by our model. More importantly, our multi-level structural loss model also provides much higher DSC values with an average 1.5% improvement compared to the RSTN model. Selective segmentation examples of thin data and thick data are provided in Fig. 8. The visual comparison can demonstrate that our model can identify the small, long, and narrow structures more accurately.

Because the 2D slices along with the coronal and sagittal views are much fewer than the slices on the axial view for the thick data resulting in the low segmentation accuracy on the two views, we use the axial view model instead of the fusion model for both our approach and the RSTN. Although the values of DSC for the three models significantly decline, our models triumphs over RSTN and nnU-Net with about a 1.5% advantage. More importantly, our multi-level structural network gives significantly better performance on the thick dataset.

5.3. MSD spleen dataset

5.3.1. Numerical analysis

Our models also work well on other organ segmentation tasks such as spleen segmentation. We implement the proposed model and RSTN along with the axial view without three-dimensional fusion because the dataset is small and varies significantly in the other two dimensions. We evaluate our approaches by comparing with two boundary-based methods and two 3D approaches, i.e., EBP model [64], LSM model [72], V-net [38] and nnU-Net[48], which are all re-implemented by ourselves for a fair comparison. Note that the ground truths are used to generate the 3-D bounding

Table 7
Segmentation accuracy comparison between our model and the other two methods on ISICDM datasets, where the results of RSTN [60] and nnU-Net [48] were re-implemented by ourselves.

Dataset	Method	Averaged DSC	Min	Max	Iteration	Time (s)
Thin	RSTN [60]	86.18% ± 5.26%	70.81%	92.35%	6.11	110.16
	nnU-Net [48]	86.71% ± 5.03%	74.09%	**94.79%**	-	1021.3
	MLL model	**87.63% ± 4.73%**	**74.53%**	93.70%	**3.92**	**81.21**
	MLN model	87.33% ± 4.91%	74.35%	93.12%	4.42	99.50
Thick	RSTN [60]	79.93% ± 7.29%	55.28%	89.95%	7.78	11.09
	nnU-Net [48]	80.75% ± 8.83%	47.00%	91.19%	-	136.4
	MLL model	82.23% ± 6.41%	**65.04%**	90.02%	6.64	**10.62**
	MLN model	**82.66%** ± 6.82%	62.13%	**91.36%**	**5.81**	11.07

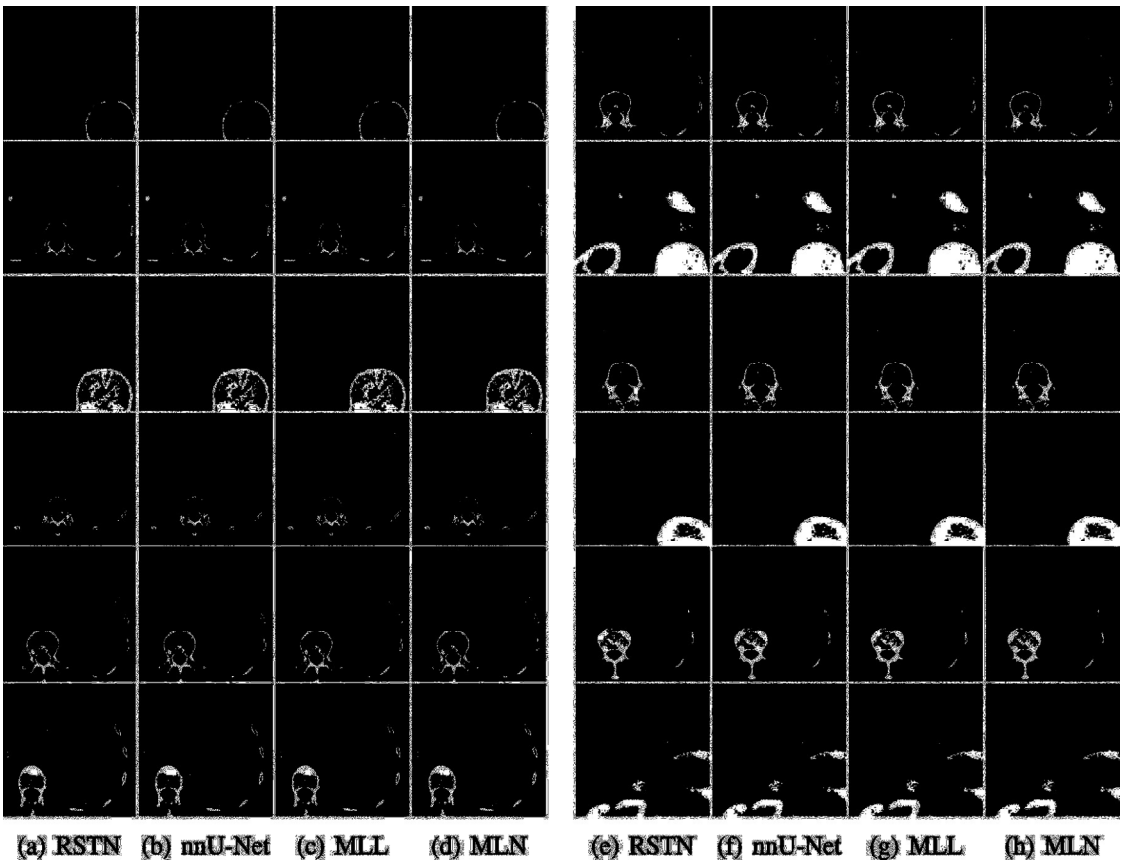


Fig. 8. Selective visual comparison between our models and the recurrent saliency transformation network [60] on ISICDM dataset, where red, green and yellow indicate the ground truth, prediction and overlapped region, respectively. The left part is the thin sub-dataset, while the right part is the thick sub-dataset. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

box for both EBP and V-net. The DSC of RSTN reported in [64] is 89.5%, which is much lower than our re-implementation due to the major voting by three axes. As shown in Table 8, our model provides the highest DSC and lowest variance, which is much better than other approaches. In addition, our models take less computational time than others except for V-Net, which used the label to crop a bounding box. Since both our models take fewer iterations to satisfy the stopping criteria, the inference time is saved. As can be seen, our multi-level structural network can further improve the averaged DSC by another 0.6%. Since the boundary of the spleen is much smoother than the boundaries of the pancreas, the boundary branch and pixel branch can help to well approximate the ground-truth, and play a strong guidance role for the regional features. We also display the selective 2D segmentation results in Fig. 9, where our model outperforms other models visually.

5.3.2. Ablation analysis of modules in MLN model

In this subsection, we explore the ablation study to evaluate the individual contribution of the pixel module and multi-level saliency guidance module. To be specific, we use three baseline models in comparison, the first one obtained by replacing the pixel module with four shared MLPs, the second one obtained by replacing the multi-level saliency guidance module by four 3×3 convolutional layers, and the last one obtained by replacing both the pixel module and the multi-level saliency guidance module. As we can see from Table 9, with similar numbers of parameters, FLOPs, and inference times, our multi-level saliency guidance module can improve DSC by 0.3% with fewer iterations. Our pixel module performs as a non-local module by extracting features from uncertain pixels, while the four shared MLPs used in PointRend [58] only catch local features. By our local features, the DSC is improved

Table 8
Segmentation accuracy (DSC %) comparison between our model and the state-of-the-arts on the MSD spleen dataset, where the results were re-implemented by ourselves.

Method	DSC	Min	Max	Iterations	Time (m)	Parameters	Flops
EBP [64]	92.79% ± 3.76%	84.81%	96.75%	10	60.26	0.2×10^{7}	0.5×10^{10}
LSM [72]	93.03% ± 2.20%	89.22%	96.29%	3	4.81	4.1×10^{7}	10.6×10^{10}
V-Net [38]	92.11% ± 7.93%	59.95%	96.53%	-	0.10	1.9×10^{7}	68.4×10^{10}
RSTN [60]	95.26% ± 1.31%	92.61%	97.29%	3.55	0.21	26.9×10^{7}	44.1×10^{10}
nnU-Net [48]	95.77% ± 1.55%	91.33%	97.75%	-	4.5	6.2×10^{7}	293.5×10^{10}
MLL model	95.96% ± 1.14%	94.18%	97.85%	**2.55**	0.18	3.1×10^{7}	14.9×10^{10}
MLN model	**96.58% ± 0.97%**	**94.87%**	**98.13%**	**2.55**	0.19	3.6×10^{7}	16.9×10^{10}

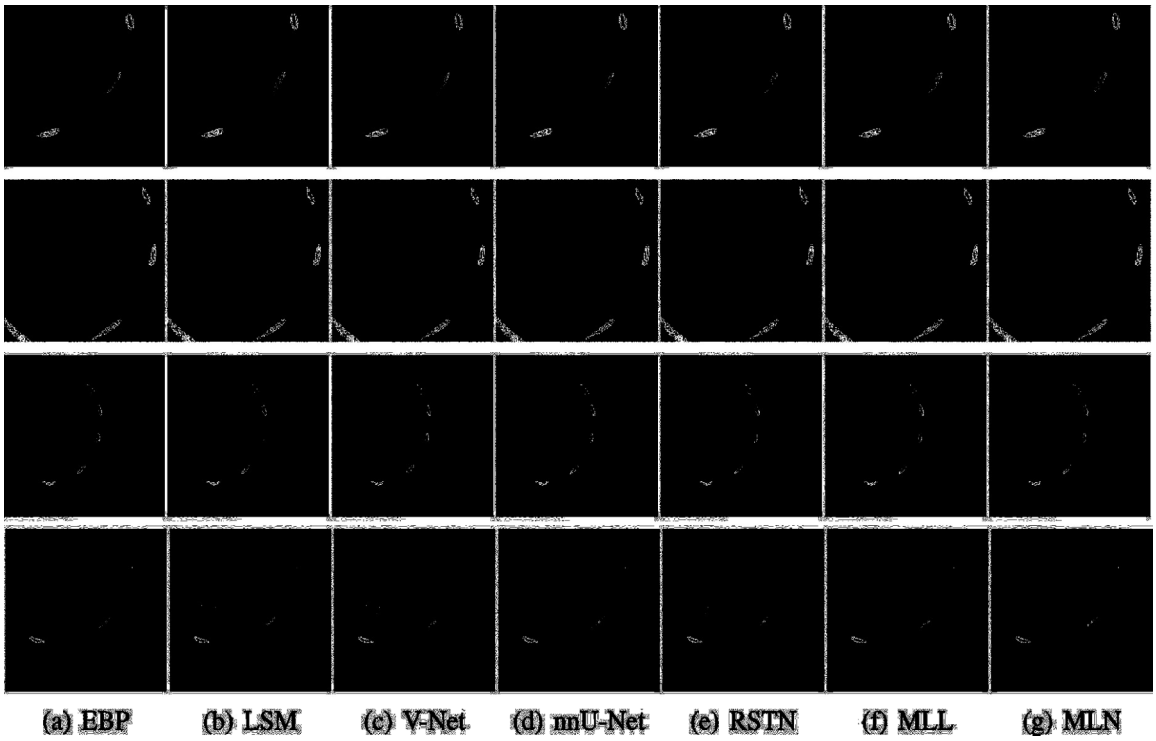


Fig. 9. Selective visual comparison between our model and the recurrent saliency transformation network on MSD spleen dataset, where red, green and yellow indicate the ground truth, prediction and overlapped region, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 9
Ablation study for the modules of our proposed multi-level structural net, where the pixel module and multi-level saliency guidance module (MLSG module) are replaced by four shared MLPs and four 3×3 convolutional layers, respectively.

Method	DSC	Min	Max	Iterations	Time	Parameters	Flops
w/o Pixel&MLSG module	96.02% ± 1.30%	93.56%	98.26%	3.90	13.3s	4.0×10^{7}	2.5×10^{11}
w/o MLSG module	96.28% ± 1.20%	93.43%	**98.18%**	2.65	11.8s	4.0×10^{7}	2.5×10^{11}
w/o Pixel module	96.35% ± 1.37%	92.12%	98.16%	**2.55**	10.9s	3.6×10^{7}	1.7×10^{11}
MLN model	**96.58% ± 0.97%**	**94.87%**	98.13%	**2.55**	11.4s	3.6×10^{7}	1.7×10^{11}

by another 0.2%. Thus, both the pixel module and the multi-level saliency guidance module contribute to the segmentation accuracy.

6. Conclusions

In this work, we promoted a novel loss function by penalizing the multi-level structural information to better aggregate multi-scale features for small organ segmentation. We adopted the coarse-to-fine atrous convolution model and took full consideration of the small size of the target by designing a proper reception field to preserve more low-level features. Comprehensive experiments on the public pancreas and spleen datasets demonstrated the superiority of the proposed method in dealing with small organ segmentation problems. Our multi-level structural loss not only consistently improved the segmentation accuracy, but also enhanced the training stability and the convergence of the fine model in the testing stage. The numerical experiments also demonstrated our multi-level structural network outperformed the single head model on the NIH pancreas dataset and the thin subset of the ISICDM dataset.

Although the proposed MLL model and MLN model exhibited advantages on both pancreas and spleen segmentation problems, the performance of the two models did not always coincide. Thus, the theoretical mechanism of multi-level structural information fusion is still an open question to be studied. One more limitation of our models is that the three-dimensional information is still

inadequate for our 2D approaches. Although we fuse three networks along the three views to provide more spatial details, the three models along different directions are independently trained without spatial correlation, which also increases the computational costs. Our future works include to develop more efficient loss functions and network structures to investigate multi-level structural information for realizing precise medical segmentation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Yueyun Liu: Software, Validation, Writing – original draft. **Yuping Duan:** Conceptualization, Methodology, Supervision, Writing – review & editing. **Tieyong Zeng:** Methodology.

Acknowledgment

The work was partially supported by National Natural Science Foundation of China (NSFC 12071345, 11701418), Major Science and Technology Project of Tianjin 18ZXRHSY00160 and Recruitment Program of Global Young Expert.

References

[1] R. Adams, L. Bischof, Seeded region growing, IEEE Trans Pattern Anal Mach Intell 16 (6) (1994) 641–647.
[2] T.H. Lee, M.F.A. Fauzi, R. Komiya, Segmentation of CT brain images using k-means and EM clustering, in: 2008 Fifth International Conference on Computer Graphics, Imaging and Visualisation, 2008, pp. 339–344.
[3] A.M. Ali, A.A. Farag, A.S. El-Baz, Graph cuts framework for kidney segmentation with prior shape constraints, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, 2007, pp. 384–392.
[4] D. Mumford, J. Shah, Optimal approximations by piecewise smooth functions and associated variational problems, Commun Pure Appl Math 42 (1989) 577–685.
[5] T.F. Chan, L.A. Vese, Active contours without edges, IEEE Trans. Image Process. 10 (2) (2001) 266–277.
[6] R. Wolz, C. Chu, K. Misawa, M. Fujiwara, K. Mori, D. Rueckert, Automated abdominal multi-organ segmentation with subject-specific atlas generation, IEEE Trans Med Imaging 32 (9) (2013) 1723–1730.
[7] E. Shelhamer, J. Long, T. Darrell, Fully convolutional networks for semantic segmentation, IEEE Trans Pattern Anal Mach Intell 39 (4) (2017) 640–651.
[8] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, 2015, pp. 234–241.
[9] L.-C. Chen, G. Papandreou, F. Schroff, H. Adam, Rethinking atrous convolution for semantic image segmentation. arXiv:1706.05587.
[10] S. Xie, Z. Tu, Holistically-nested edge detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1395–1403.
[11] J. Wu, W. Zhou, T. Luo, L. Yu, J. Lei, Multiscale multilevel context and multimodal fusion for RGB-d salient object detection, Signal Processing 178 (2021) 107766.
[12] Y. Pang, T. Wang, R.M. Anwer, F.S. Khan, L. Shao, Efficient featurized image pyramid network for single shot detector, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 7336–7344.
[13] S.W. Zamir, A. Arora, S. Khan, M. Hayat, F.S. Khan, M.-H. Yang, L. Shao, Learning enriched features for real image restoration and enhancement, in: European Conference on Computer Vision, 2020, pp. 492–511.
[14] H. Chen, X. Qi, L. Yu, P.-A. Heng, DCAN: Deep contour-aware networks for accurate gland segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2487–2496.
[15] H. Shen, R. Wang, J. Zhang, S.J. McKenna, Boundary-aware fully convolutional network for brain tumor segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, 2017, pp. 433–441.
[16] Y. Xu, Y. Li, Y. Wang, M. Liu, Y. Fan, M. Lai, I. Eric, C. Chang, Gland instance segmentation using deep multichannel neural networks, IEEE Trans. Biomed. Eng. 64 (12) (2017) 2901–2912.
[17] J. Duan, J. Schlemper, W. Bai, T.J.W. Dawes, G. Bello, G. Doumou, A. De Marvao, D.P. O'Regan, D. Rueckert, Deep nested level sets: Fully automated segmentation of cardiac MR images in patients with pulmonary hypertension, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, 2018, pp. 595–603.
[18] Y. Pang, Y. Li, J. Shen, L. Shao, Towards bridging semantic gap to improve semantic segmentation, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 4230–4239.
[19] Z. Zhang, Y. Pang, Cgnet: cross-guidance network for semantic segmentation, Science China Information Sciences 63 (2) (2020) 1–16.
[20] F. Fang, J. Li, Y. Yuan, T. Zeng, G. Zhang, Multilevel edge features guided network for image denoising, IEEE Trans Neural Netw Learn Syst 32 (9) (2021) 3956–3970.
[21] Y. Fang, T. Zeng, Learning deep edge prior for image denoising, Comput. Vision Image Understanding 200 (2020) 103044.
[22] F. Fang, J. Li, T. Zeng, Soft-edge assisted network for single image super-resolution, IEEE Trans. Image Process. 29 (2020) 4656–4668.
[23] W. Zhou, J. Liu, J. Lei, L. Yu, J.-N. Hwang, Gmnet: Graded-feature multilabel-learning network for RGB-thermal urban scene semantic segmentation, IEEE Trans. Image Process. 30 (2021) 7790–7802.
[24] W. Zhou, Y. Zhu, J. Lei, J. Wan, L. Yu, Ccafnet: Crossflow and cross-scale adaptive fusion network for detecting salient objects in rgb-d images, IEEE Trans Multimedia (2021). 1–1
[25] F. Jia, J. Liu, X.-C. Tai, A regularized convolutional neural network for semantic image segmentation, Analysis and Applications 19 (01) (2020) 147–165.
[26] F. Jia, X.-C. Tai, J. Liu, Nonlocal regularized CNN for image segmentation, Inverse Problems & Imaging 14 (5) (2020) 891–911.
[27] Y. Yang, Q. Zhong, Y. Duan, T. Zeng, A weighted bounded hessian variational model for image labeling and segmentation, Signal Processing 173 (2020) 107564.
[28] T. Wu, X. Gu, Y. Wang, T. Zeng, Adaptive total variation based image segmentation with semi-proximal alternating minimization, Signal Processing 183 (2021) 108017.
[29] P. Hu, B. Shuai, J. Liu, G. Wang, Deep level sets for salient object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2300–2309.
[30] D. Marcos, D. Tuia, B. Kellenberger, L. Zhang, M. Bai, R. Liao, R. Urtasun, Learning deep structured active contours end-to-end, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8877–8885.
[31] X. Chen, B.M. Williams, S.R. Vallabhaneni, G. Czanner, R. Williams, Y. Zheng, Learning active contour models for medical image segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 11624–11632.
[32] Y. Kim, S. Kim, T. Kim, C. Kim, CNN-based semantic segmentation using level set loss, in: 2019 IEEE Winter Conference on Applications of Computer Vision, 2019, pp. 1752–1760.
[33] A. Hatamizadeh, A. Hoogi, D. Sengupta, W. Lu, B. Wilcox, D. Rubin, D. Terzopoulos, Deep active lesion segmentation, in: International Workshop on Machine Learning in Medical Imaging, 2019, pp. 98–105.
[34] M. Zhang, B. Dong, Q. Li, Deep active contour network for medical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, 2020, pp. 321–331.
[35] T.F. Chan, S. Esedoglu, M. Nikolova, Algorithms for finding global minimizers of image segmentation and denoising models, SIAM J Appl Math 66 (5) (2006) 1632–1648.
[36] B. Kim, J.C. Ye, Mumford–shah loss functional for image segmentation with deep learning, IEEE Trans. Image Process. 29 (2019) 1856–1866.
[37] J. Ma, J. He, X. Yang, Learning geodesic active contours for embedding object global information in segmentation CNNs, IEEE Trans Med Imaging 40 (1) (2020) 93–104.
[38] F. Milletari, N. Navab, S.-A. Ahmadi, V-net: Fully convolutional neural networks for volumetric medical image segmentation, in: 2016 Fourth International Conference on 3D Vision, 2016, pp. 565–571.
[39] M.A. Rahman, Y. Wang, Optimizing Intersection-over-union in Deep Neural Networks for Image Segmentation, in: International symposium on visual computing, 2016, pp. 234–244.
[40] S.S.M. Salehi, D. Erdogmus, A. Gholipour, Tversky loss function for image segmentation using 3Dfully convolutional deep networks, in: International Workshop on Machine Learning in Medical Imaging, 2017, pp. 379–387.
[41] C.H. Sudre, W. Li, T. Vercauteren, S. Ourselin, M.J. Cardoso, Generalised Dice Overlap as a Deep Learning Loss Function for Highly Unbalanced Segmentations, in: Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support, 2017, pp. 240–248.
[42] N. Abraham, N.M. Khan, A novel focal tversky loss function with improved attention U-Net for lesion segmentation, in: 2019 IEEE 16th International Symposium on Biomedical Imaging, 2019, pp. 683–687.
[43] S.R. Hashemi, S.S.M. Salehi, D. Erdogmus, S.P. Prabhu, S.K. Warfield, A. Gholipour, Asymmetric loss functions and deep densely-connected networks for highly-imbalanced medical image segmentation: application to multiple sclerosis lesion detection, IEEE Access 7 (2019) 1721–1735.
[44] S. Yang, J. Kweon, Y.-H. Kim, Major vessel segmentation on x-ray coronary angiography using deep networks with a novel penalty loss function, in: International Conference on Medical Imaging with Deep Learning – Extended Abstract Track, 2019.
[45] Z. Wu, C. Shen, H.A. van den, Bridging category-level and instance-level semantic image segmentation. arXiv preprint arXiv:1605.06885.
[46] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollar, Focal loss for dense object detection, IEEE Trans Pattern Anal Mach Intell 42 (2) (2020) 318–327.

[47] F. Caliva, C. Iriondo, A.M. Martinez, S. Majumdar, V. Pedoia, Distance map loss penalty term for semantic segmentation, in: International Conference on Medical Imaging with Deep Learning – Extended Abstract, 2019.
[48] F. Isensee, P.F. Jaeger, S.A.A. Kohl, J. Petersen, K.H. Maier-Hein, nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation, Nat. Methods 18 (2) (2021) 203–211.
[49] W. Zhu, Y. Huang, L. Zeng, X. Chen, Y. Liu, Z. Qian, N. Du, W. Fan, X. Xie, Anatomynet: deep learning for fast and fully automated whole-volume segmentation of head and neck anatomy, Med Phys 46 (2) (2018) 576–589.
[50] D. Karimi, S.E. Salcudean, Reducing the hausdorff distance in medical image segmentation with convolutional neural networks, IEEE Trans Med Imaging 39 (2) (2020) 499–513.
[51] H. Kervadec, J. Bouchtiba, C. Desrosiers, E. Granger, J. Dolz, I.B. Ayed, Boundary loss for highly unbalanced segmentation, in: Proceedings of The 2nd International Conference on Medical Imaging with Deep Learning, 2019, pp. 285–296.
[52] Y. Xue, H. Tang, Z. Qiao, G. Gong, Y. Yin, Z. Qian, C. Huang, W. Fan, X. Huang, Shape-aware organ segmentation by predicting signed distance maps, Proceedings of the AAAI Conference on Artificial Intelligence 34 (07) (2020) 12565–12572.
[53] Y. Lan, Y. Xiang, L. Zhang, An elastic interaction-based loss function for medical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, 2020, pp. 755–764.
[54] J.H. Moltz, A. Hansch, B. Lassen-Schmidt, B. Haas, A. Genghi, J. Schreier, T. Morgas, J. Klein, Learning a loss function for segmentation: A feasibility study, in: 2020 IEEE 17th International Symposium on Biomedical Imaging, 2020, pp. 957–960.
[55] C. Azzopardi, K.P. Camilleri, Y.A. Hicks, Bimodal automated carotid ultrasound segmentation using geometrically constrained deep neural networks, IEEE J Biomed Health Inform 24 (4) (2020) 1004–1015.
[56] H.R. Roth, L. Lu, A. Farag, H.-C. Shin, J. Liu, E.B. Turkbey, R.M. Summers, Deeporgan: Multi-level deep convolutional networks for automated pancreas segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, 2015, pp. 556–564.
[57] A. Bansal, X. Chen, B. Russell, A. Gupta, D. Ramanan, Pixelnet: towwards a general pixel-level architecture. arXiv:1609.06694.
[58] A. Kirillov, Y. Wu, K. He, R. Girshick, Pointrend: Image segmentation as rendering, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2020, pp. 9799–9808.
[59] Y. Zhou, L. Xie, W. Shen, Y. Wang, E.K. Fishman, A.L. Yuille, A fixed-point model for pancreas segmentation in abdominal CT scans, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, 2017, pp. 693–701.
[60] Q. Yu, L. Xie, Y. Wang, Y. Zhou, E.K. Fishman, A.L. Yuille, Recurrent saliency transformation network: Incorporating multi-stage visual cues for small organ segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 641–647.
[61] J. Ma, Z. Wei, Y. Zhang, Y. Wang, R. Lv, C. Zhu, C. Gaoxiang, J. Liu, C. Peng, L. Wang, Y. Wang, J. Chen, How distance transform maps boost segmentation CNNs: An empirical study, in: Proceedings of Machine Learning Research, volume 121, 2020, pp. 479–492.
[62] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, u. Kaiser, I. Polosukhin, Attention is all you need, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017, pp. 6000–6010.
[63] L. Xie, Q. Yu, Y. Zhou, Y. Wang, E.K. Fishman, A.L. Yuille, Recurrent saliency transformation network for tiny target segmentation in abdominal CT scans, IEEE Trans Med Imaging 39 (2) (2020) 514–525.
[64] T. Ni, L. Xie, H. Zheng, E.K. Fishman, A.L. Yuille, Elastic boundary projection for 3d medical image segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 2109–2118.
[65] Y. Liu, S. Liu, U-Net for pancreas segmentation in abdominal CT scans, ISBI Challenge (2018).
[66] F. Li, W. Li, Y. Shu, S. Qin, B. Xiao, Z. Zhan, Multiscale receptive field based on residual network for pancreas segmentation in CT images, Biomed Signal Process Control 57 (2020) 101828.
[67] G. Zeng, G. Zheng, Holistic decomposition convolution for effective semantic segmentation of medical volume images, Med Image Anal 57 (2019) 149–164.
[68] Z. Zhu, Y. Xia, W. Shen, E. Fishman, A. Yuille, A 3D coarse-to-fine framework for volumetric medical image segmentation, in: 2018 International Conference on 3D Vision, 2018, pp. 682–690.
[69] Y. Xia, L. Xie, F. Liu, Z. Zhu, E.K. Fishman, A.L. Yuille, Bridging the gap between 2d and 3d organ segmentation with volumetric fusion net, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, 2018, pp. 445–453.
[70] H. Chen, X. Wang, Y. Huang, X. Wu, Y. Yu, L. Wang, Harnessing 2d networks and 3d features for automated pancreas segmentation from volumetric CT images, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, 2019, pp. 339–347.
[71] P. Hu, X. Li, Y. Tian, T. Tang, T. Zhou, X. Bai, S. Zhu, T. Liang, J. Li, Automatic pancreas segmentation in CT images with distance-based saliency-aware denseASPP network, IEEE J Biomed Health Inform 25 (5) (2020) 1601–1611.
[72] L. Guo, Y. Liu, Y. Wang, Y. Duan, X.-C. Tai, Learned snakes for 3D image segmentation, Signal Processing 183 (2021) 108013.