Graph Cuts Loss to Boost Model Accuracy and Generalizability for Medical Image Segmentation

Zhou Zheng[1] Masahiro Oda[1] Kensaku Mori[1,2,*]
[1]Nagoya University [2]National Institute of Informatics
{zzheng, moda}@mori.m.is.nagoya-u.ac.jp, kensaku@is.nagoya-u.ac.jp

Abstract

Segmentation accuracy and generalization ability are essential for deep learning models, especially in medical image segmentation. We present a novel, robust yet straightforward loss function to boost model accuracy and generalizability for medical image segmentation. We reformulate the graph cuts cost function to a loss function for supervised learning. The graph cuts loss innately focuses on a dual penalty to optimize the regional properties and boundary regularization. We benchmark the proposed loss on six public retinal vessel segmentation datasets with a comprehensive intra-dataset and cross-dataset evaluation. Results reveal that the proposed loss is more generalizable, narrowing the performance gap between different architectures. Besides, models trained with our loss show higher segmentation accuracy and better generalization ability than those trained with other mainstream losses. Moreover, we extend our loss to other segmentation tasks, e.g., left atrium and liver tumor segmentation. The proposed loss still achieves comparable performance to the state-of-the-art, demonstrating its potential for any N-D segmentation problem. The code is available at https://github.com/zzhenggit/graph_cuts_loss.

1. Introduction

Deep learning models, especially the convolutional neural networks (CNNs), have made remarkable progress in medical image segmentation [20]. CNN training typically relies on a loss function to calculate errors in these predictions and ground truth. Regional loss functions, such as cross-entropy (CE) [27], or Dice Coefficient (DC) [24], or the combination of CE and DC [6], are commonly used in various approaches by evaluating the pixel-wise similarities. However, they still succumb to poor accuracy in complex medical images. For instance, the retinal vessel appears as the thin elongated structure with variation in width and length. Only relying on pixel-level affinity, models easily result in disconnected and missing segmentation [19]. As alternatives, a number of novel losses [9, 19, 8, 14, 13, 1, 29] emerged, promoting model accuracy for various challenging segmentation tasks. In addition to accuracy, generalizability is another essential ability for models, allowing accurate and robust segmentation for cross datasets. Several works argue that CNN architecture optimization [15] and tricks of data normalization and augmentation [5] benefit model generalizability. However, the effect of loss on model generalizability is rarely explored. To our knowledge, existing losses typically only focus on model accuracy but hardly ever investigate their roles in model generalization ability.

Active-contour-based approaches [9, 19, 8] implemented the variational energy functional as a loss function for supervised learning. Their extra-introduced geometrical constraints are useful for medical images with complex structures, e.g., tubular/curvilinear structures. However, a drawback of most active-contour-based losses is their instability in early training steps with random initialization [19]. In addition to the variational method, e.g., the active contour model, another type of energy-based segmentation approach is the combinatorial model, which is optimized by a cost function defined on a discrete set of variables [2]. One typical representative is the graph cuts algorithm [3], which also allows the unification of boundary cues, region cues, and topological constraints as the active contour model. Classical graph-cuts-based methods have shown their remarkable potentials in challenging segmentation tasks, e.g, vessel segmentation [28, 32, 12]. In this paper, we ask and answer the following research questions: since works of [17, 16, 9, 19, 8] combine the superiorities of classical active contour models and modern learning-based models by reformulating the curve-evolving energy functional to a loss function, can we introduce the graph cuts cost function to a loss function to integrate the advantages of classical graph cuts methods and the learning-based models? Besides, will the proposed loss do help to improve model accuracy and generalizability?

*Corresponding author

This paper pioneers implementing the graph cuts cost function as a loss function and exploring the role of the proposed loss in both model segmentation accuracy and model generalization ability for medical image segmentation. To learn the graph cuts (GC) loss function that suits the supervision framework, we assume the graph is constructed based on the probability prediction, and we also redefine each edge cost within the graph. The presented GC loss function combines boundary regularization with region-based properties in the same fashion as the graph cuts cost function. We extensively evaluate the proposed loss function on six public retinal vessel segmentation datasets through intra-dataset and cross-dataset validation. Results indicate that our loss is more generalizable than other mainstream losses and can endow models with higher accuracy and better generalizability. Furthermore, additional experiments on other segmentation tasks, e.g., left atrium and liver tumor segmentation, show the GC loss can be readily applied to other N-D segmentation problems.

2. Related Work

Loss for medical image segmentation. CE and DC are widely used regional losses for pixel-wise classification by measuring the region similarities between probability prediction and corresponding ground truth. Since region-based losses may not yield meaningful segmentation for complex medical images, a number of losses are newly proposed for various segmentation tasks. For instance, Chen et al. [9] presented an active contour loss with an additional term of contour length, as pixel-wise losses would lead to a noisy result with many contours in the background. Some similar works also introduced active contour methods to losses [8, 17, 16, 19]. In [17] and [16], Mumford-Shah functional was reformulated to a loss function, but its application to biomedical segmentation was not investigated. Compared to [9], the approach of [8] added one more curvature term as the geometrical constraint to improve performance, and it was applicable to preserving the connectedness and reducing the missing segmentation for tabular structures. However, this loss introduced more than one hyperparameter and was sensitive to the hyperparameters. Different to [9, 8] that were based on the Chan-Vese model [4], method of [19] was derived from the elastic interaction model [33]. It was more stable during training and also showed advantages in tackling the segmentation of vascular structures. In addition, there are also some boundary-based losses. Kervadec et al. [14] proposed a boundary loss (BD) for highly unbalanced segmentation, trying to solve the problem that regional losses would perform poorly when the size of the target foreground region is much less than the background size. Hausdorff Distance (HD) loss [13] is another edge-based loss proposed for reducing the metric of HD. Another recently proposed clDice [29] focused on the topology for tubular structure segmentation. It showed better topology-preserving than the DC loss, but it was not compared with other existing losses. More losses for medical image segmentation can be found in [22].

Edge	Weight	For
$\{u, v\}$	$B_{\{u,v\}}$	$u, v \in N$
$\{s, 0\}$	$\lambda \cdot R(A_s = 1)$	$s \in S$
$\{s, 1\}$	$\lambda \cdot R(A_s = 0)$	$s \in S$

Table 1. Cost of every edge within the graph in our work.

Graph cuts. We have witnessed the success of the graph cuts algorithm in the past decades, the theory of which is well suited for segmentation, where an undirected graph $G = \langle V, E \rangle$ is usually employed. G is defined as a set of nodes V and a set of undirected edges E that connect these nodes, and there are two particular nodes called terminals in G. E consists of two types of undirected edges: n-links (neighboring links) and t-links (terminal links), and each e of E has a nonnegative cost. A cut is a subset of edges $C \subset E$ such that the background terminal and object terminal could be separated on the included graph $G(C) = \langle V, E \setminus C \rangle$. The dedicated pioneer application of graph cuts to segmentation should be the work of Boykov et al. [3]. Given a 2D (or 3D) image I, let $S = (s_1, s_2, \ldots, s_n)$ denote the set of n pixels (or voxels) in I, and $N = (\{u, v\}_1, \{u, v\}_2, \ldots, \{u, v\}_\eta)$ indicate the set of η unpaired pairs of neighboring pixels (or voxels) under a standard 8-neighborhood (or 26-neighborhood) system. In a binary segmentation task, each pixel (or voxel) s_i with value p_i in S would be assigned to label 1 (object) or label 0 (background). Let A_i denote the assignment of each pixel (or voxel) s_i, and then there would be a one-to-one correspondence between a vector $A = (A_1, A_2, \ldots, A_n)$ and a segmentation result. The graph cuts cost $E(A)$ consists of a regional term $\lambda \cdot R(A) = \lambda \cdot \sum_{s \in S} R_s(A_s)$, where λ is a nonnegative coefficient, and a boundary term $B(A) = \sum_{\{u,v\} \in N} B_{\{u,v\}} \cdot \delta(A_u, A_v)$, where $\delta(A_u, A_v)$ equals 1 if $A_u \neq A_v$ otherwise 0.

3. Graph Cuts Loss Function

To learn the graph cuts (GC) loss function that suits the supervised learning, unlike [3], we assume the graph G is based on the probability prediction I. An example of graph G for a 2D 3×3 probability prediction I and its segmentation in our study is shown in Fig. 1. To be specific, each pixel s has two t-links $\{s, 0\}$ and $\{s, 1\}$ respectively connecting it to background terminal (label 0) and object terminal (label 1), and each pair of neighboring pixels of $\{u, v\}$ is connected by an n-link. In our work, we redefine the costs of edges in G, and details are shown in Table 1. In addition, the cost $R(A)$ of each t-link is defined in Eq. (1). This definition is proper because pixel value p_i in prediction al-

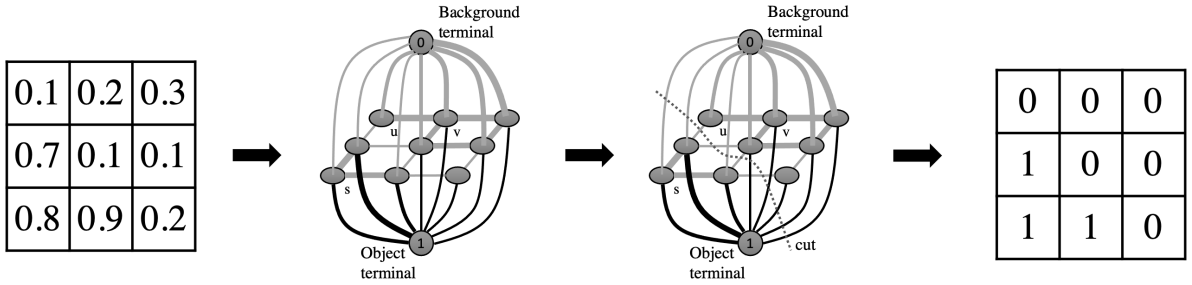

Figure 1. An example of graph G for a 2D 3×3 probability prediction I and its segmentation. The cost of every edge is reflected by the thickness of the edge. The regional terms in Eq. (1) define the costs of t-links. The boundary term in Eq. (2) defines the costs of n-links.

ready shows the probability of assigning s_i to object (label 1). And the cost $B_{\{u,v\}}$ of each n-link is defined in Eq. (2), in which $dist(u,v)$ measures the distance between paired pixels, and if $|p_u - p_v|$ is small, $B_{\{u,v\}}$ would be large, so this function penalizes a lot for discontinuities. A feasible cut C in our work should serve exactly one t-link at each s and include $\{u,v\}$ if and only if u, v are t-linked to different terminals.

$$R_s(A_s = 1) = -\log p_i \quad R_s(A_s = 0) = -\log(1 - p_i) \tag{1}$$

$$B_{\{u,v\}} \propto \exp(-\frac{(p_u - p_v)^2}{2\sigma^2}) \cdot \frac{1}{dist(u,v)} \tag{2}$$

Due to the supervision framework in our study, for a given probability prediction I, we already know the ground truth and the corresponding cut C. For instance, assume we own the truth label as illustrated in Fig. 1(d), then in order to get this expected segmentation, we would strictly require the cut C in Fig. 1(c) to cut the graph. Generally, given an $n \times n$ ground truth T and its corresponding specific cut $\tilde{C}$, let $M = (m_1, m_2, \ldots, m_K)$ be a set of all K possible $n \times n$ probability prediction images of T, and $F = (C_1, C_2, \ldots, C_L)$ be a set of all L feasible cuts of each m_i. Thus, each m_i to yield T would cost energy $E_i(A(\tilde{C}))$. Besides, we can immediately get that if and only if m_i equals the ground truth T, then $R(A) = 0$ and $|p_u - p_v| = 1$ as $\delta(A_u, A_v) = 1$ in $B(A)$, such that energy $E_i(A(\tilde{C}))$ would be minimum. Thus, by minimizing $E_i(A(\tilde{C}))$, we would get m_i that is closest to the ground truth. Let y_i denote each pixel value of ground truth T, and then we get the GC loss function $Loss_{GC}$:

$$Loss_{GC} = \lambda \cdot R(A) + B(A) \tag{3}$$

where

$$R(A) = -\sum_{i=1}^{n} [y_i \cdot \log p_i + (1 - y_i) \cdot \log(1 - p_i)] \tag{4}$$

$$B(A) \propto \sum_{\{u,v\} \in N} \exp(-\frac{(p_u - p_v)^2}{2\sigma^2}) \cdot \frac{1}{dist(u,v)} \cdot \delta(y_u, y_v) \tag{5}$$

and

$$\delta(y_u, y_v) = \begin{cases} 1, & if \quad y_u \neq y_v \\ 0, & otherwise. \end{cases} \tag{6}$$

There are two coefficients (λ and σ) in the combinatorial framework. The coefficient λ in Eq. (3) controls the balance between the region penalty $R(A)$ and the boundary penalty $B(A)$, while the parameter σ in Eq. (5) can be estimated as 'camera noise' [3]. As introducing more than one parameter would make models sensitive to parameter setting, it would also be more difficult to search an optimal group of parameters than a single optimal parameter. Thus, to eliminate σ, we make an approximation for the boundary term as

$$B(A) = \sum_{\{u,v\} \in N} \{1 - |p_u - p_v|\} \cdot \delta(y_u, y_v) \tag{7}$$

By observing the form of the GC loss function, we can find its region term $R(A)$ in Eq. (4) is equal to binary cross-entropy (BCE). That is to say, through learning from the graph cuts cost, the GC loss has one more boundary term $B(A)$ than CE. It is worth mentioning that the method of [7] for instance segmentation similarly combined the region and boundary penalties. However, its strategy was ad hoc and was realized by extracting the region and edge simultaneously via two different branches in a multi-task setting. By contrast, our loss is derived from a cost function and can be integrated into any semantic segmentation model. Due to the different approaches and utilization, comparison with this method is out of the scope of this paper.

4. Experiments

This section elaborates the following experiments: we first performed a thorough intra-dataset and cross-dataset

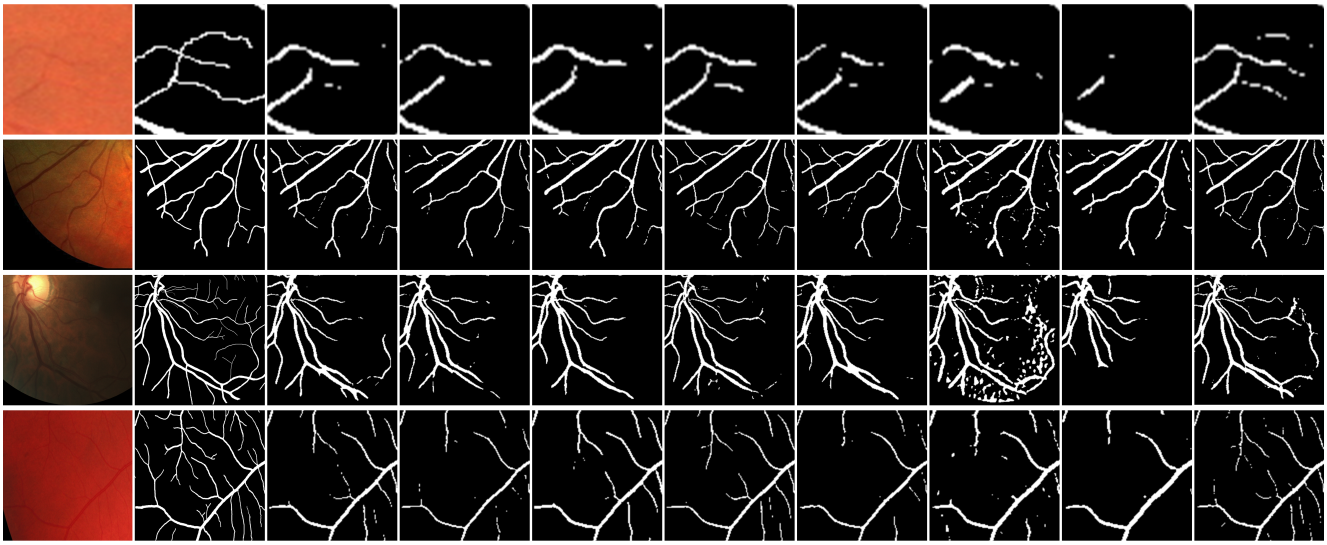

Figure 2. Visual comparisons. From top to bottom, we show the results of intra-dataset validation on the combined SDC dataset, the results of cross-dataset validation on the IOSTAR dataset, the LES-AV dataset, and the HRF dataset. All predictions are obtained using U-Net. It can be noted that our loss can better detect small retinal vessels than other mainstream losses.

evaluation for the proposed GC loss on the retinal vessel segmentation task. Then we extended the GC loss to 3D left atrium and liver tumor segmentation to explore its applicability to other organs and structures. All experiments relied on the Pytorch platform and an NVIDIA 1080ti GPU.

4.1. Retinal Vessel Segmentation

Datasets and metrics. We combined the STARE dataset [11], the DRIVE dataset [31][1] and the CHASEDB1 dataset [26][2] to produce a more general dataset (SDC) with more samples for intra-dataset validation, inspired by [5]. The formed SDC dataset comprised 88 images and was randomly split into 58/15/15 for intra-dataset training, validation, and test. We then directly fed the IOSTAR dataset [35], the HRF dataset [18], and the LES-AV dataset [25] to the previously trained models for cross-dataset evaluation. The IOSTAR dataset comprises 30 fundus images with a resolution of 1024 $\times$ 1024 pixels. The HRF dataset consists of 45 fundus images with a resolution of 3504 $\times$ 2336 pixels. The LES-AV dataset combines 21 typical fundus images with a resolution of 1620 $\times$ 1444 pixels and one pathological image with a resolution of 1958 $\times$ 2196 pixels. We utilized seven different metrics, including Dice, the 95th percentile of Hausdorff Distance (HD95), the newly introduced pixel-wise measure clDice [29] for tubular structure segmentation, Specificity, Sensitivity, Accuracy, and AUC.

Implemention details. We compared the proposed GC loss with other state-of-the-art losses: CE, DC, the pixel-wise clDice loss [29] for topology-preserving, the active-contour-like losses (the elastic interaction-based loss (EIB) [19], the active contour loss (AC) [9], and the active contour loss with Euler's Elastica (ACE) [8][3]), the boundary-based losses (the HD loss [13] and the BD loss [14]). Due to the training instability of AC, HD, and BD, we combined them with DC via a rebalanced-increasing-parameter training strategy, following [14]. We chose U-Net [27] and pre-trained VGG-16 [30] based FCN [21] as our segmentation backbones in this part of the experiment in order to investigate the loss generalizability to different architectures with either training from scratch or fine-tuning. During the experiment, all images were resized to the resolution of 448 $\times$ 448 pixels. Data augmentation of image flipping, scaling, shifting, and color jittering [10] were utilized. All models were trained with the Adam optimizer, the batch size 8, and the maximum epoch 600. For a fair comparison, we searched the optimal learning rate in $[10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}]$ for each loss function as [13, 8]. The learning rate was divided by 10 when the validation performance did not improve over 20 epochs. The training process was early stopped when the learning rate was smaller than 10^{-7}.

Ablation study. λ in Eq. (3) controls the balance between the region and boundary properties. There would be only a boundary term contributing to the constraints when λ is 0, and GC would be approximately the same as CE when λ grows to infinite. Based on the SDC dataset and the back-

[1] We used the first annotations.
[2] We used the first annotations.
[3] As the ACE loss was quite sensitive to parameter setting in our experiment, we do not report its results in this paper for a fair comparison.

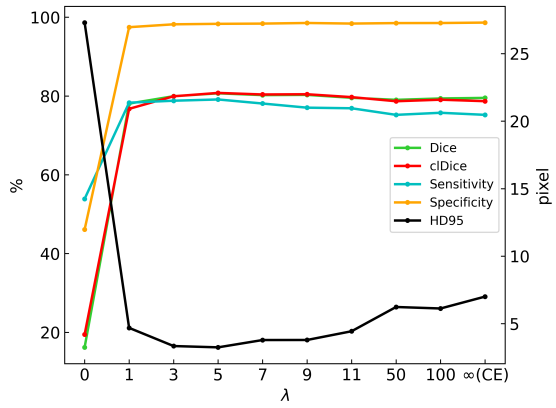

Figure 3. Ablation study on different λ, using U-Net and the SDC dataset with five metrics. By considering overall metrics, the optimal λ of GC for retinal vessel segmentation ranges from 3 to 9.

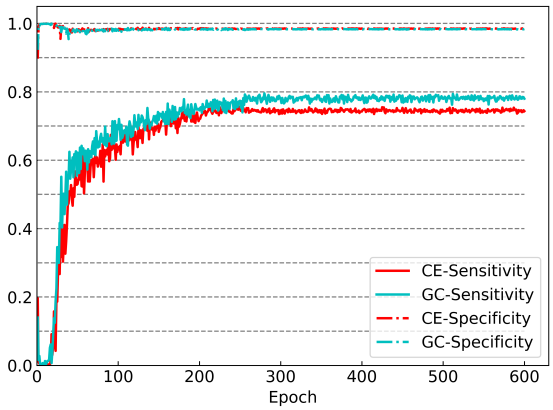

Figure 4. Comparison of Sensitivity and Specificity obtained by CE and GC on the validation set during training. The backbone is U-Net. GC majorly improves Sensitivity compared to CE but slightly influences Specificity in the meantime.

bone U-Net, we ran several ablations to analyze the effect of different λ on the loss performance, as shown in Fig. 3, where five metrics are reported. We can observe that the worst result occurs due to the lack of region information. The GC loss starts to achieve meaningful results when λ is larger than 0. It can be found that Dice, clDice, and HD95 reach relatively stable and optimal when λ ranges from 3 to 9. In addition, we can reveal the advantage of the combinatorial framework by focusing on the changes in Sensitivity and Specificity. The values of Specificity show a slight and continuous increase, whereas the values of Sensitivity decrease in a more obvious way, with the increment of λ from 1 to infinite. Coupled with the analysis of visual examples of CE and GC in Fig. 2, we can conclude that adding the boundary regularization benefits the detection of small vessel areas as it improves Sensitivity via increasing true positives. Nevertheless, it may also influence Specificity as it may introduce more false positives. Fortunately, results tell us that the advantage outweighs the disadvantage, as GC (λ = 5) only owns 0.3% lower Specificity but 4% higher Sensitivity than CE. To this end, by considering overall metrics, the optimal λ of GC for retinal vessel segmentation approximately ranges from 3 to 9.

Intra-dataset validation. Intra-dataset validation results are illustrated in Table 2, where we report results of other mainstream losses and our loss with λ equaling 5, 7, and 9.

First, let us focus on CE and GC. A major observation could be that GC guides models to obtain higher Sensitivity than CE, which further confirms the conclusion drawn from the ablation study. To be specific, when λ is set as 5, U-Net-GC improves approximately 4% in Sensitivity and owns only 0.3% lower Specificity than U-Net-CE. Similarly, FCN-GC achieves about 3% higher Sensitivity and only 0.4% lower Specificity than FCN-CE. A detailed comparison of Sensitivity and Specificity obtained by CE and GC on the validation set during training is shown in Fig. 4. We can note that GC majorly improves Sensitivity compared to CE but slightly influences Specificity.

Then let us pay attention to model segmentation accuracy guided by these losses. DC yields higher Sensitivity than CE for both U-Net and FCN. The clDice loss obtains the highest clDice scores for both two nets, demonstrating its efficiency for topology-preserving. However, results reflect a limitation that the clDice loss may overlook other metrics. EIB and AC are both active-contour-based losses, but their performance is quite different for this dataset. To be specific, EIB gets considerably higher scores of Dice, clDice, and Sensitivity and lower HD95 values than AC, but its AUC score is much lower than that of AC. The gap between their performance should be attributed to the different baseline approaches, as AC was derived from the region-based model [4] while EIB was inspired by the elastic-interaction-based approach [33]. HD and BD are both boundary-based losses. Though U-Net-HD and FCN-BD respectively bring the highest Sensitivity, their overall performance is not as good as ours. By contrast, it can be observed that for both two nets, our loss achieves the best scores of Dice and HD95 and also reaches the best or the second-best scores of clDice, Accuracy, and AUC among all losses.

Moreover, there are also apparent gaps between the results of the two nets trained with the same loss. For example, FCN-DC achieves about 4% higher Sensitivity than U-Net-DC. FCN-CE brings about 2% higher Sensitivity than U-Net-DC. The Dice score of FCN-clDice is about 5% lower than that of U-Net-clDice. The AUC score of FCN-EIB is about 4% lower than that of U-Net-EIB. FCN-

Dataset	Network	Loss	Dice ↑	HD95 ↓	clDice ↑	Specificity ↑	Sensitivity ↑	Accuracy ↑	AUC ↑
SDC	U-Net	DC	79.94 (1.02)	5.28 (1.07)	80.03 (1.51)	98.26 (0.23)	78.52 (3.09)	96.42 (0.08)	**97.84 (0.22)**
		CE	79.52 (0.94)	7.00 (1.94)	78.68 (1.56)	**98.64 (0.05)**	75.22 (1.83)	96.47 (0.12)	97.73 (0.37)
		clDice	80.03 (0.20)	4.98 (0.34)	**81.60 (0.20)**	97.68 (0.13)	**82.52 (0.67)**	96.25 (0.07)	96.93 (0.20)
		EIB	79.61 (0.57)	**3.52 (0.20)**	80.70 (0.47)	98.62 (0.08)	75.54 (1.25)	96.48 (0.08)	90.97 (1.41)
		AC	76.08 (2.72)	7.77 (2.65)	76.01 (2.53)	**98.69 (0.20)**	69.69 (5.10)	96.01 (0.33)	96.31 (1.12)
		HD	76.77 (1.28)	5.78 (0.98)	78.48 (1.42)	96.54 (0.16)	**83.98 (1.34)**	95.35 (0.27)	97.22 (0.44)
		BD	76.48 (1.11)	11.07 (0.80)	75.85 (0.42)	98.17 (0.65)	73.63 (2.34)	95.88 (0.38)	95.89 (0.45)
		GC ($\lambda = 5$)	**80.68 (0.32)**	**3.25 (0.17)**	**80.79 (0.39)**	98.33 (0.07)	79.12 (0.09)	**96.55 (0.07)**	97.80 (0.24)
		GC ($\lambda = 7$)	80.23 (0.54)	3.79 (0.40)	80.39 (0.77)	98.38 (0.15)	78.09 (1.54)	96.50 (0.08)	97.83 (0.15)
		GC ($\lambda = 9$)	**80.29 (0.27)**	3.79 (0.30)	80.45 (0.29)	98.54 (0.18)	77.04 (1.62)	**96.56 (0.02)**	**97.99 (0.18)**
	FCN	DC	79.03 (0.70)	4.56 (0.21)	80.44 (0.49)	97.34 (0.27)	82.81 (1.06)	95.98 (0.19)	97.19 (0.24)
		CE	**80.72 (0.36)**	4.45 (0.14)	80.20 (0.24)	**98.55 (0.01)**	77.63 (0.52)	**96.62 (0.05)**	**98.28 (0.04)**
		clDice	75.28 (0.66)	5.09 (0.21)	**80.68 (0.22)**	95.61 (0.33)	**86.91 (0.74)**	94.78 (0.23)	93.50 (0.33)
		EIB	78.47 (0.10)	3.82 (0.05)	79.58 (0.10)	**98.51 (0.08)**	74.48 (0.47)	96.29 (0.03)	87.33 (0.88)
		AC	77.04 (0.29)	5.21 (0.48)	79.06 (0.99)	96.57 (0.46)	84.14 (2.48)	95.41 (0.20)	96.79 (0.10)
		HD	78.49 (0.36)	4.48 (0.23)	79.93 (0.23)	97.16 (0.25)	83.03 (1.03)	95.84 (0.14)	96.64 (0.47)
		BD	68.57 (1.70)	4.98 (0.18)	78.43 (0.11)	92.86 (0.79)	**89.29 (0.77)**	92.50 (0.65)	94.89 (0.18)
		GC ($\lambda = 5$)	80.70 (0.15)	**3.01 (0.08)**	80.07 (0.01)	98.16 (0.08)	80.22 (0.74)	96.50 (0.00)	98.14 (0.04)
		GC ($\lambda = 7$)	80.43 (0.36)	3.19 (0.12)	80.06 (0.35)	98.32 (0.10)	78.70 (1.17)	96.51 (0.03)	98.07 (0.11)
		GC ($\lambda = 9$)	**80.78 (0.23)**	**3.19 (0.04)**	**80.51 (0.17)**	98.33 (0.04)	79.16 (0.57)	**96.57 (0.03)**	**98.21 (0.03)**

Table 2. Intra-dataset validation on the generated SDC dataset, where the best and second best results are marked in **bold** **red** and **black**. We report the average (standard deviation) results based on three runs.

Dataset	Network	Loss	Dice ↑	HD95 ↓	clDice ↑	Specificity ↑	Sensitivity ↑	Accuracy ↑	AUC ↑
IOSTAR	U-Net	DC	76.26 (0.67)	6.48 (0.38)	82.28 (0.48)	97.81 (0.14)	77.85 (0.23)	96.22 (0.14)	**97.65 (0.08)**
		CE	75.34 (0.95)	8.04 (0.76)	80.81 (0.98)	98.39 (0.08)	72.27 (1.91)	96.31 (0.09)	**97.61 (0.21)**
		clDice	76.20 (0.24)	6.39 (0.41)	**83.58 (0.43)**	97.32 (0.09)	**81.23 (1.07)**	96.04 (0.01)	96.67 (0.24)
		EIB	**77.13 (0.20)**	5.37 (0.24)	**82.55 (0.36)**	**98.40 (0.19)**	74.96 (1.38)	**96.53 (0.07)**	91.18 (1.21)
		AC	73.65 (1.06)	7.80 (1.21)	80.05 (0.97)	**98.43 (0.29)**	69.45 (3.53)	96.13 (0.03)	96.13 (0.65)
		HD	70.86 (2.36)	9.56 (2.39)	77.99 (2.81)	95.35 (0.70)	**85.03 (0.43)**	94.53 (0.64)	96.99 (0.42)
		BD	72.02 (2.38)	12.96 (0.90)	78.09 (1.37)	97.41 (0.72)	73.65 (1.77)	95.52 (0.58)	95.43 (0.76)
		GC ($\lambda = 5$)	76.95 (0.48)	**5.19 (0.38)**	81.98 (0.50)	98.22 (0.14)	76.04 (0.45)	**96.45 (0.11)**	97.11 (0.27)
		GC ($\lambda = 7$)	**77.19 (0.31)**	**5.17 (0.30)**	82.45 (0.20)	98.11 (0.33)	77.17 (2.88)	96.44 (0.08)	97.45 (0.22)
		GC ($\lambda = 9$)	76.05 (0.23)	5.49 (0.12)	81.73 (0.19)	98.35 (0.16)	73.65 (1.44)	96.38 (0.05)	97.48 (0.33)
	FCN	DC	75.89 (0.71)	5.84 (0.37)	**82.66 (0.39)**	97.17 (0.35)	81.75 (1.46)	95.94 (0.22)	97.29 (0.27)
		CE	76.17 (0.32)	6.27 (0.10)	81.34 (0.28)	**98.54 (0.16)**	72.44 (1.64)	**96.46 (0.02)**	**98.24 (0.01)**
		clDice	72.61 (0.61)	5.79 (0.32)	**83.19 (0.19)**	95.59 (0.20)	**86.75 (0.38)**	94.89 (0.17)	93.64 (0.17)
		EIB	76.04 (0.13)	5.30 (0.27)	81.79 (0.47)	**98.39 (0.21)**	73.31 (1.71)	96.39 (0.06)	86.96 (0.93)
		AC	73.58 (0.53)	5.89 (0.50)	81.13 (1.10)	96.56 (0.33)	82.05 (2.78)	95.40 (0.12)	96.73 (0.13)
		HD	75.39 (0.61)	5.62 (0.22)	81.91 (0.23)	97.18 (0.30)	80.90 (1.12)	95.88 (0.19)	96.67 (0.60)
		BD	65.86 (2.04)	5.91 (0.25)	79.58 (0.52)	93.10 (1.02)	**88.88 (1.71)**	92.77 (0.81)	95.30 (0.20)
		GC ($\lambda = 7$)	**76.56 (0.58)**	5.53 (0.34)	80.73 (0.48)	98.17 (0.24)	75.74 (2.58)	96.38 (0.04)	97.88 (0.13)
		GC ($\lambda = 7$)	**76.43 (0.49)**	**5.27 (0.06)**	81.29 (0.22)	98.26 (0.19)	74.86 (1.92)	96.40 (0.06)	97.86 (0.15)
		GC ($\lambda = 9$)	76.29 (0.19)	**5.21 (0.22)**	81.33 (0.09)	98.38 (0.15)	73.76 (1.12)	**96.42 (0.05)**	**97.97 (0.04)**

Table 3. Cross-dataset validation on the IOSTAR dataset, where the best and second best results are marked in **bold** **red** and **black**. We report the average (standard deviation) results based on three runs.

AC even achieves about 14% higher Sensitivity than that of U-Net-AC. FCN-HD owns approximately 2% higher Dice score than U-Net-HD. U-Net-BD yields about 8% higher Dice than FCN-BD. The above evidence indicates that other losses are easy to get turbulent results when integrated into different networks. On the contrary, the GC loss guides U-Net and FCN to yield much closer metric values, where all the percentage differences are within about 1%.

This intra-dataset evaluation reveals that our loss is more generalizable to different architectures and further boosts model accuracy in most metrics than other losses.

Cross-dataset validation. To explore whether the proposed loss could promote model generalizability, we further conducted three cross-dataset experiments. We directly employed the models trained with the utilized losses to predict the IOSTAR dataset, the HRF dataset, and the LES-AV dataset without any operation.

Validation results of the IOSTAR dataset are illustrated in Table 3. Interestingly, we can note that, for this dataset, the models trained with different losses achieve similar rankings to the SDC dataset. For instance, U-Net and FCN trained with our loss still achieve the best scores regarding Dice and HD95 and the second-best performance in some other metrics. The two nets integrated with the clDice loss also bring the highest clDice scores. In addition, FCN-CE still obtains the highest Specificity, Accuracy, and AUC. Almost all models do not show apparent degrade and upgrade in rankings on this dataset compared to the SDC dataset. This phenomenon may be attributed to the similar data distributions between the two datasets.

Major performance turbulences on the LES-AV dataset and the HRF dataset could be observed in Table 4 and Table 5. One of the apparent changes is that the models trained with GC beat the models guided by the clDice loss to achieve the highest clDice score on the two datasets. It may demonstrate that, compared to the clDice loss, the GC loss

Dataset	Network	Loss	Dice ↑	HD95 ↓	clDice ↑	Specificity ↑	Sensitivity ↑	Accuracy ↑	AUC ↑
LES-AV	U-Net	DC	78.09 (1.53)	20.00 (4.03)	77.52 (1.96)	98.62 (0.01)	76.94 (2.23)	97.20 (0.15)	97.29 (0.37)
		CE	77.53 (1.54)	22.68 (5.22)	76.36 (2.42)	98.87 (0.32)	73.82 (4.51)	97.23 (0.14)	97.31 (0.46)
		clDice	76.92 (0.94)	22.61 (4.05)	78.00 (1.28)	98.21 (0.01)	78.79 (1.36)	96.95 (0.10)	94.62 (1.35)
		EIB	**80.22 (1.45)**	**11.13 (2.37)**	**80.07 (1.80)**	**99.09 (0.14)**	76.09 (3.17)	**97.58 (0.13)**	92.73 (1.41)
		AC	72.30 (5.50)	28.60 (7.39)	71.49 (5.13)	**98.97 (0.22)**	65.79 (9.03)	96.82 (0.38)	93.45 (2.72)
		HD	71.27 (2.56)	15.98 (3.23)	74.10 (2.53)	96.80 (0.36)	80.65 (1.94)	95.75 (0.44)	96.46 (0.64)
		BD	72.36 (2.31)	36.74 (9.66)	70.78 (3.06)	98.63 (0.36)	68.38 (4.50)	96.65 (0.27)	92.29 (1.86)
		GC ($\lambda = 5$)	**80.90 (0.72)**	**8.37 (0.87)**	**81.42 (0.67)**	98.56 (0.24)	**82.14 (1.34)**	**97.48 (0.15)**	**97.98 (0.06)**
		GC ($\lambda = 7$)	79.66 (0.62)	13.44 (0.41)	79.76 (0.57)	98.68 (0.09)	**78.91 (0.32)**	97.38 (0.09)	**97.70 (0.23)**
		GC ($\lambda = 9$)	78.82 (0.35)	17.02 (1.54)	78.41 (0.43)	98.75 (0.15)	76.93 (0.91)	97.33 (0.08)	97.05 (0.81)
	FCN	DC	78.05 (0.80)	8.89 (0.18)	81.05 (0.31)	97.78 (0.19)	84.28 (0.50)	96.90 (0.16)	97.65 (0.33)
		CE	80.98 (0.25)	10.66 (0.25)	81.19 (0.17)	98.72 (0.11)	80.61 (0.71)	97.54 (0.05)	98.55 (0.04)
		clDice	73.41 (0.72)	9.54 (1.23)	81.49 (0.52)	96.39 (0.23)	**87.83 (0.73)**	95.83 (0.18)	94.06 (0.28)
		EIB	80.53 (0.47)	7.47 (0.14)	80.86 (0.29)	**98.95 (0.09)**	77.77 (0.51)	**97.55 (0.07)**	88.76 (0.88)
		AC	74.83 (0.34)	9.93 (1.58)	79.86 (1.16)	97.01 (0.32)	85.15 (2.10)	96.24 (0.16)	96.80 (0.13)
		HD	77.08 (0.84)	8.31 (0.81)	80.53 (0.61)	97.56 (0.26)	84.58 (1.12)	96.71 (0.19)	97.29 (0.41)
		BD	66.53 (2.06)	9.70 (0.42)	79.63 (0.37)	94.53 (0.65)	**88.32 (0.47)**	94.14 (0.57)	95.11 (0.21)
		GC ($\lambda = 5$)	**81.52 (0.25)**	**5.52 (0.72)**	**82.05 (0.30)**	98.55 (0.04)	83.16 (0.49)	97.54 (0.03)	**98.59 (0.09)**
		GC ($\lambda = 7$)	**81.48 (0.14)**	**6.58 (0.21)**	81.60 (0.37)	**98.72 (0.12)**	81.45 (1.20)	**97.59 (0.04)**	98.47 (0.08)
		GC ($\lambda = 9$)	81.24 (0.20)	7.36 (0.57)	**81.86 (0.51)**	98.61 (0.11)	82.07 (1.03)	97.54 (0.05)	**98.56 (0.10)**

Table 4. Cross-dataset validation on the LES-AV dataset, where the best and second best results are marked in **bold** **red** and **black**. We report the average (standard deviation) results based on three runs.

Dataset	Network	Loss	Dice ↑	HD95 ↓	clDice ↑	Specificity ↑	Sensitivity ↑	Accuracy ↑	AUC ↑
HRF	U-Net	DC	69.82 (0.71)	7.73 (1.78)	74.68 (1.67)	96.05 (0.61)	79.64 (2.94)	94.79 (0.35)	95.87 (0.37)
		CE	71.10 (0.56)	9.70 (2.54)	73.49 (2.21)	**96.87 (0.40)**	76.54 (3.16)	95.30 (0.15)	95.52 (0.58)
		clDice	67.59 (0.87)	6.79 (0.79)	75.94 (0.51)	94.96 (0.36)	82.51 (1.04)	94.00 (0.28)	94.27 (0.60)
		EIB	**73.92 (0.29)**	**4.26 (0.22)**	**77.15 (0.93)**	96.61 (0.09)	**83.21 (0.97)**	**95.58 (0.04)**	94.67 (0.55)
		AC	71.38 (0.21)	10.60 (2.95)	71.61 (2.68)	**97.50 (0.71)**	72.79 (5.07)	**95.59 (0.26)**	93.19 (1.90)
		HD	63.12 (1.56)	7.17 (1.10)	72.94 (1.53)	93.70 (0.34)	81.56 (1.41)	92.76 (0.39)	94.83 (0.38)
		BD	65.09 (3.66)	13.89 (1.52)	70.20 (1.03)	95.70 (1.34)	73.41 (1.85)	93.98 (1.12)	91.25 (0.47)
		GC ($\lambda = 5$)	72.70 (0.34)	**3.85 (0.33)**	**77.18 (0.37)**	96.14 (0.09)	**84.28 (0.61)**	95.23 (0.07)	**96.95 (0.08)**
		GC ($\lambda = 7$)	72.04 (0.61)	4.91 (0.96)	76.35 (0.56)	96.23 (0.20)	82.48 (1.52)	95.17 (0.13)	**96.63 (0.23)**
		GC ($\lambda = 9$)	**72.94 (0.75)**	6.02 (0.68)	75.78 (0.58)	96.77 (0.18)	80.33 (0.75)	95.50 (0.16)	96.45 (0.27)
	FCN	DC	66.06 (1.18)	6.19 (0.27)	73.04 (0.70)	94.80 (0.50)	80.76 (0.93)	93.71 (0.40)	94.65 (0.12)
		CE	71.84 (0.39)	6.67 (0.38)	74.35 (0.13)	**96.76 (0.19)**	78.52 (0.65)	**95.35 (0.13)**	96.23 (0.09)
		clDice	59.75 (0.79)	6.19 (0.30)	72.52 (0.27)	92.14 (0.39)	**83.33 (0.48)**	91.46 (0.32)	90.37 (0.36)
		EIB	71.68 (0.37)	5.44 (0.17)	73.64 (0.55)	**96.87 (0.08)**	77.56 (0.83)	**95.37 (0.05)**	90.58 (0.68)
		AC	63.93 (0.52)	6.66 (0.81)	70.97 (1.36)	94.27 (0.55)	79.88 (2.30)	93.16 (0.33)	93.21 (0.35)
		HD	65.28 (1.32)	5.93 (0.30)	72.35 (0.48)	94.58 (0.49)	80.59 (0.49)	93.50 (0.42)	93.96 (0.36)
		BD	51.84 (1.85)	5.57 (0.11)	68.11 (0.41)	88.24 (1.09)	**84.70 (0.49)**	87.98 (0.97)	89.80 (0.30)
		GC ($\lambda = 5$)	**72.35 (0.16)**	**3.83 (0.17)**	**75.05 (0.24)**	96.30 (0.15)	82.53 (0.74)	95.23 (0.08)	**96.43 (0.10)**
		GC ($\lambda = 7$)	72.19 (0.08)	**4.13 (0.09)**	74.83 (0.35)	96.44 (0.08)	81.30 (0.44)	95.27 (0.04)	96.33 (0.08)
		GC ($\lambda = 9$)	**72.45 (0.27)**	4.39 (0.11)	**75.21 (0.46)**	96.51 (0.04)	81.24 (0.23)	95.33 (0.05)	**96.44 (0.11)**

Table 5. Cross-dataset validation on the HRF dataset, where the best and second best results are marked in **bold** **red** and **black**. We report the average (standard deviation) results based on three runs.

potentially guides models to learn a more generalized retinal vessel representation and better preserve topology for unseen datasets. Moreover, the two nets trained with our loss also reach the best or the second-best scores in most other metrics. Besides, the models trained with EIB realize comparable segmentation accuracy to ours in terms of some metrics. Specifically, EIB brings competitive Dice, clDice, Specificity, and Accuracy scores when integrated into U-Net and FCN for both two datasets. However, it could not achieve the same accuracy in HD95, Sensitivity, and AUC as the GC loss. In addition, U-Net and FCN trained with our loss still obtain closer results on these two datasets.

This cross-dataset analysis reflects that our loss can further benefit model generalization ability than other losses.

Qualitative comparison. Fig. 2 shows some examples for visual comparisons. It can be observed that the DC loss achieves more visually acceptable results than the CE loss for these cases. The segmented vessels of the clDice loss show better connectivity than that of other losses. The EIB loss performs better than the AC loss, especially in detecting small vessels. The HD loss tends to bring more false positives than the other losses, while the BD loss yields the fewest true positives. It is clear to see that the proposed GC loss can better detect small retinal vessels than other losses.

4.2. Other Segmentation Tasks

Datasets. To evaluate the applicability of the proposed loss to other segmentation tasks, we further extended the GC loss to 3D with the consideration of a 26-neighborhood system. We respectively evaluated our loss on the left atrium (LA) segmentation challenge dataset[4] that comprises 100 3D MR training cases, and the liver tumor segmentation (LiTs) challenge dataset[5] including 118 training scans.

[4] https://atriaseg2018.cardiacatlas.org/

[5] https://competitions.codalab.org/competitions/17094

Implemention details. We chose V-Net [24] as the 3D segmentation backbone in this part of the experiment. For the LA dataset, we followed the implementation of [34, 23] to employ heart-centered crop and z-score normalization for all cases. We respectively trained the models with 16 and 80 samples and then evaluated them with 20 samples to explore the robustness of the proposed loss to different training sizes. For the LiTs dataset, we also pre-processed the scans with liver-centered crop and z-score normalization and split the set into 90/28 for training and validation as [23]. We empirically set λ to 5 for the GC loss following the previous 2D retinal vessel segmentation.

Validation results. Experiment results are described in Table 6, where we also report results of some previous methods using various losses, such as BD, HD, and the signed distance function (SDF) loss.

We can note that GC is robust to the different training sizes of the LA dataset. When trained with 16 samples, GC achieves the highest Dice score and the second-lowest HD95 value among all approaches. Similarly, GC still obtains better results than CE in Dice and HD95, and it also gets a higher Dice score than DC and the combination of DC and CE with 80 training scans. As the training size increases, the gap among loss performance is narrowed.

When evaluated on the LiTs dataset, we found a limitation of GC, and let us discuss it. As mentioned above, GC is a combinatorial framework including a region term, i.e., CE, and a boundary regularization term. It inherits the advantage of the boundary term but also the disadvantage of CE. Concretely, it cannot also well handle the unbalanced segmentation as CE. As the liver tumors characterize unbalanced distributions, CE and GC can not efficiently play their roles when the tumor areas are much smaller than the background. To address this problem, we tried the trick of applying another region indicator for GC, and it worked. Precisely, similar to HD and BD, we also used the rebalanced-increasing-parameter training strategy to combine DC with GC. This strategy enabled GC to achieve the highest Dice score than other approaches, as illustrated in Table 6. As we can see, combining DC and GC obtains about 1.6% higher Dice score than the combination of DC and CE and improves about 5.3% Dice compared to the single DC loss.

Dataset	Method	Dice ↑	HD95 ↓
LA (80)	DC + CE [34]	91.14	5.75
	DC*	91.43 (0.17)	**5.39 (0.40)**
	CE*	91.02 (0.18)	6.25 (1.26)
	GC ($\lambda = 5$)*	**91.68 (0.39)**	5.98 (1.97)
LA (16)	DC + CE [23]	84.4	20.1
	BD [23]	85.0	20.8
	HD [23]	85.5	15.9
	SDF [23]	84.2	**13.5**
	M† + SDF + L1 + L2 [23]	84.5	24.7
	R† + SDF + L1 + L2 [23]	85.1	16.7
	CE*	82.79 (2.48)	15.28 (0.67)
	DC*	84.31 (0.68)	14.93 (0.54)
	GC ($\lambda = 5$)*	**86.66 (0.61)**	13.51 (2.97)
LiTs	DC + CE [23]	51.0	43.6
	BD [23]	52.5	26.3
	HD [23]	52.0	28.8
	SDF [23]	47.6	31.1
	M† + SDF + L1 [23]	48.1	31.5
	M† + SDF + L2 [23]	47.1	**25.5**
	R† + SDF + L1 [23]	48.4	32.2
	R† + SDF + L2 [23]	48.6	31.0
	DC*	47.33 (0.35)	34.46 (3.94)
	DC + GC ($\lambda = 5$)*	**52.64 (0.81)**	40.76 (1.33)

Table 6. Quantitative validation on the LA dataset and the LiTs dataset using V-Net-like architectures. The best results are marked in **bold black**. M† and R† mean the Multi-heads and Rec-branch architectures in [23], respectively. * indicates the reimplemented methods in our study. We report the average (standard deviation) results based on three runs.

5. Computational Efficiency

Our approach requires $O(n^2)$ computational complexity for a 2D $n \times n$ image. During our experiment, the training time for a batch size of 8 and a resolution of 448 $\times$ 448 pixels was about 0.24s for GC, which was very close to the clDice loss (0.22s) and AC (0.18s) and faster than HD (0.45s) and BD (0.46s) that was without pre-computed level-set function but was slower than CE (0.07s) and DC (0.09s). Thus, the computation cost of our loss is in the medium compared with the mainstream losses. Since the inference time is naturally not associated with the loss, we do not report it here.

6. Conclusion

In this paper, we proposed a novel graph cuts (GC) loss function to promote model segmentation accuracy and generalization ability for medical image segmentation, inspired by the combinatorial graph cuts cost function. The GC loss innately comprised the region and boundary penalties. We pioneered exploring the role of the proposed loss in both model segmentation accuracy and model generalization ability via the retinal vessel segmentation task. Compared to the state-of-the-art, the GC loss was more generalizable to model architectures and further boosted model accuracy and generalizability. Furthermore, we extended the GC loss to 3D left atrium and liver tumor segmentation to show that it could be applied to any N-D segmentation problem. In addition to providing a competitive alternative loss to enrich the loss repository, we hope our approach could inspire the work related to model generalizability.

Acknowledgement: This work was supported by JST CREST Grant Number JPMJCR20D5, Japan.

References

[1] N. Abraham and 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 (ISBI 2019)*, pages 683–687, 2019. 1

[2] Yuri Boykov and Gareth Funka-Lea. Graph cuts and efficient ND image segmentation. *International journal of computer vision*, 70(2):109–131, 2006. 1

[3] Y.Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary amp; region segmentation of objects in N-D images. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 1, pages 105–112 vol.1, 2001. 1, 2, 3

[4] T.F. Chan and L.A. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, 2001. 2, 5

[5] Chen Chen, Wenjia Bai, Rhodri H. Davies, Anish N. Bhuva, Charlotte H. Manisty, Joao B. Augusto, James C Moon, Nay Aung, Aaron M. Lee, Mihir M. Sanghvi, Kenneth Fung, Jose Miguel Paiva, Steffen E. Petersen, Elena Lukaschuk, Stefan K. Piechnik, Stefan Neubauer, and Daniel Rueckert. Improving the generalizability of convolutional neural network-based segmentation on CMR images. *Frontiers in Cardiovascular Medicine*, 7:105, 2020. 1, 4

[6] Cheng Chen, Qi Dou, Yueming Jin, Hao Chen, Jing Qin, and Pheng-Ann Heng. Robust multimodal brain tumor segmentation via feature disentanglement and gated fusion. In Dinggang Shen, Tianming Liu, Terry M. Peters, Lawrence H. Staib, Caroline Essert, Sean Zhou, Pew-Thian Yap, and Ali Khan, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, pages 447–456, Cham, 2019. Springer International Publishing. 1

[7] Hao Chen, Xiaojuan Qi, Lequan Yu, and Pheng-Ann Heng. DCAN: Deep contour-aware networks for accurate gland segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 3

[8] Xu Chen, Xiangde Luo, Yitian Zhao, Shaoting Zhang, Guotai Wang, and Yalin Zheng. Learning Euler's elastica model for medical image segmentation. *arXiv preprint arXiv:2011.00526*, 2020. 1, 2, 4

[9] Xu Chen, Bryan M. Williams, Srinivasa R. Vallabhaneni, Gabriela Czanner, Rachel Williams, and Yalin Zheng. Learning active contour models for medical image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 2, 4

[10] Z. Gu, J. Cheng, H. Fu, K. Zhou, H. Hao, Y. Zhao, T. Zhang, S. Gao, and J. Liu. CE-Net: Context encoder network for 2D medical image segmentation. *IEEE Transactions on Medical Imaging*, 38(10):2281–2292, 2019. 4

[11] A.D. Hoover, V. Kouznetsova, and M. Goldbaum. Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response. *IEEE Transactions on Medical Imaging*, 19(3):203–210, 2000. 4

[12] Daniel Jimenez-Carretero, David Bermejo-Peláez, Pietro Nardelli, Patricia Fraga, Eduardo Fraile, Raúl San José Estépar, and Maria J Ledesma-Carbayo. A graph-cut approach for pulmonary artery-vein segmentation in noncontrast CT images. *Medical Image Analysis*, 52:144–159, 2019. 1

[13] Davood Karimi and Septimiu E. Salcudean. Reducing the Hausdorff Distance in medical image segmentation with convolutional neural networks. *IEEE Transactions on Medical Imaging*, 39(2):499–513, 2020. 1, 2, 4

[14] Hoel Kervadec, Jihene Bouchtiba, Christian Desrosiers, Eric Granger, Jose Dolz, and Ismail Ben Ayed. Boundary loss for highly unbalanced segmentation. In M. Jorge Cardoso, Aasa Feragen, Ben Glocker, Ender Konukoglu, Ipek Oguz, Gozde Unal, and Tom Vercauteren, editors, *Proceedings of The 2nd International Conference on Medical Imaging with Deep Learning*, volume 102 of *Proceedings of Machine Learning Research*, pages 285–296, London, United Kingdom, 08–10 Jul 2019. PMLR. 1, 2, 4

[15] Mahendra Khened, Varghese Alex Kollerathu, and Ganapathy Krishnamurthi. Fully convolutional multi-scale residual DenseNets for cardiac segmentation and automated cardiac diagnosis using ensemble of classifiers. *Medical Image Analysis*, 51:21–45, 2019. 1

[16] Boah Kim and Jong Chul Ye. Mumford–Shah loss functional for image segmentation with deep learning. *IEEE Transactions on Image Processing*, 29:1856–1866, 2020. 1, 2

[17] Youngeun Kim, Seunghyeon Kim, Taekyung Kim, and Changick Kim. CNN-based semantic segmentation using level set loss. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1752–1760, 2019. 1, 2

[18] Thomas Köhler, Attila Budai, Martin F. Kraus, Jan Odstrčilik, Georg Michelson, and Joachim Hornegger. Automatic no-reference quality assessment for retinal fundus images using vessel segmentation. In *Proceedings of the 26th IEEE International Symposium on Computer-Based Medical Systems*, pages 95–100, 2013. 4

[19] Yuan Lan, Yang Xiang, and Luchan Zhang. An elastic interaction-based loss function for medical image segmentation. In Anne L. Martel, Purang Abolmaesumi, Danail Stoyanov, Diana Mateus, Maria A. Zuluaga, S. Kevin Zhou, Daniel Racoceanu, and Leo Joskowicz, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*, pages 755–764, Cham, 2020. Springer International Publishing. 1, 2, 4

[20] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen A.W.M. van der Laak, Bram van Ginneken, and Clara I. Sánchez. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60 – 88, 2017. 1

[21] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 4

[22] Jun Ma, Jianan Chen, Matthew Ng, Rui Huang, Yu Li, Chen Li, Xiaoping Yang, and Anne L. Martel. Loss odyssey in medical image segmentation. *Medical Image Analysis*, 71:102035, 2021. 2

[23] Jun Ma, Zhan Wei, Yiwen Zhang, Yixin Wang, Rongfei Lv, Cheng Zhu, Chen Gaoxiang, Jianan Liu, Chao Peng, Lei Wang, Yunpeng Wang, and Jianan Chen. How distance transform maps boost segmentation CNNs: An empirical study. In Tal Arbel, Ismail Ben Ayed, Marleen de Bruijne, Maxime Descoteaux, Herve Lombaert, and Christopher Pal, editors, *Proceedings of the Third Conference on Medical Imaging with Deep Learning*, volume 121 of *Proceedings of Machine Learning Research*, pages 479–492. PMLR, 06–08 Jul 2020. 8

[24] F. Milletari, N. Navab, and S. Ahmadi. V-Net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571, 2016. 1, 7

[25] José Ignacio Orlando, João Barbosa Breda, Karel Van Keer, Matthew B. Blaschko, Pablo J. Blanco, and Carlos A. Bulant. Towards a glaucoma risk index based on simulated hemodynamics from fundus images. In Alejandro F. Frangi, Julia A. Schnabel, Christos Davatzikos, Carlos Alberola-López, and Gabor Fichtinger, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, pages 65–73, Cham, 2018. Springer International Publishing. 4

[26] Christopher G Owen, Alicja R Rudnicka, Robert Mullen, Sarah A Barman, Dorothy Monekosso, Peter H Whincup, Jeffrey Ng, and Carl Paterson. Measuring retinal vessel tortuosity in 10-year-old children: validation of the computer-assisted image analysis of the retina (caiar) program. *Investigative ophthalmology & visual science*, 50(5):2004–2010, 2009. 4

[27] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. 1, 4

[28] Ana G. Salazar-Gonzalez, Yongmin Li, and Xiaohui Liu. Retinal blood vessel segmentation via graph cut. In *2010 11th International Conference on Control Automation Robotics Vision*, pages 225–230, 2010. 1

[29] Suprosanna Shit, Johannes C. Paetzold, Anjany Sekuboyina, Ivan Ezhov, Alexander Unger, Andrey Zhylka, Josien P. W. Pluim, Ulrich Bauer, and Bjoern H. Menze. clDice - a novel topology-preserving loss function for tubular structure segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16560–16569, June 2021. 1, 2, 4

[30] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 4

[31] J. Staal, M.D. Abramoff, M. Niemeijer, M.A. Viergever, and B. van Ginneken. Ridge-based vessel segmentation in color images of the retina. *IEEE Transactions on Medical Imaging*, 23(4):501–509, 2004. 4

[32] Chenglong Wang, Masahiro Oda, Yuichiro Hayashi, Yasushi Yoshino, Tokunori Yamamoto, Alejandro F. Frangi, and Kensaku Mori. Tensor-cut: A tensor-based graph-cut blood vessel segmentation method and its application to renal artery segmentation. *Medical Image Analysis*, 60:101623, 2020. 1

[33] Yang Xiang, A.C.S. Chung, and Jian Ye. A new active contour method based on elastic interaction. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 452–457 vol. 1, 2005. 2, 5

[34] Lequan Yu, Shujun Wang, Xiaomeng Li, Chi-Wing Fu, and Pheng-Ann Heng. Uncertainty-aware self-ensembling model for semi-supervised 3D left atrium segmentation. In Dinggang Shen, Tianming Liu, Terry M. Peters, Lawrence H. Staib, Caroline Essert, Sean Zhou, Pew-Thian Yap, and Ali Khan, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, pages 605–613, Cham, 2019. Springer International Publishing. 8

[35] Jiong Zhang, Behdad Dashtbozorg, Erik Bekkers, Josien P. W. Pluim, Remco Duits, and Bart M. ter Haar Romeny. Robust retinal vessel segmentation via locally adaptive derivative frames in orientation scores. *IEEE Transactions on Medical Imaging*, 35(12):2631–2644, 2016. 4