Regularized U-Net for Fast Medical Image Segmentation

Qiaoqiao Ding
Shanghai Jiao Tong Univeristy
Shanghai
matding@nus.edu.sg

Yuanhong Jiang
Shanghai Jiao Tong Univeristy
Shanghai
william_jiang@sjtu.edu.cn

Jiulong Liu
Chinese Academy of Sciences
Beijing
jiulongliu@lsec.cc.ac.cn

Xiaoqun Zhang
Shanghai Jiao Tong Univeristy
Shanghai
xqzahng@sjtu.edu.cn

Abstract

Automatic segmentation of abdominal anatomy on computed tomography (CT) images can provide support to diagnosis, treatment planning, and treatment delivery workflows. Nowadays, CNN has become a powerful tool for image segmentation. We applied a regularized Unet for multi-organ segmentation task. We set the original image as input to predict multi-label mask.

1. Introduction

Organ segmentation is a crucially important step for computer-assisted diagnostic. In the radiation therapies plan making, segmentations of treatment volumes is also an important step.

Manual segmentation of 3D abdominal images is labor-intensive and impractical for most clinical workflows. Thus, (semi-)automated segmentation tools have been developed. But there are still mang challenges for multi-organ segementation. The first challenge comes from the diversity of the dataset, which including multi-center, multi-phase, multi-vendor, and multi-disease cases. Well generalization of the applied method is required. The second challenge is different organs have some similar structures in an image patch. The third challenge comes from the efficiency requirement for the proposed solutions.

There are many works have been proposed for multi-organ segmentation. Such as statistical models (SM) [1, 2], multi-atlas label fusion (MALF) [2, 3, 4, 5, 6] and registration-free methods[7, 8, 9]. In recent years, the deep learning based method include Vnet and nnUnet have cause much attention [10, 11].

We employed the regularized Unet (RUnet) for multi-organ segmentation. The RUnet is trained with 3D image patches. The network architecture is simple and we only modify the last activate layer, i.e. softmax layer. We replace the conventional softmax with a regularized softmax, which consider the pix-wise constant property of the maskes. There are several benefits of RUnet. Firstly, the network architecture is the same as Unet which is simple and the downsample layer decrease the network parameters. Secondly, the regularized softmax have not been increased the parameters and the traning process is fast.Thirdly, the segmentation results are improved compared with tradiontional Unet.

2. Method

2.1. Proposed Method

Let $I \in \mathcal{R}^{D\times H\times W}$ be an image with size $D \times H \times W$. Taking I as an input of a pixel-wise segmentation neural network. The network can be write as $\mathcal{N}_\Theta$ with parameter Θ, and $I^K = \mathcal{N}_\Theta(I)$. The network can be expand as

$$\begin{cases} I^0 = I \\ I^k = \mathcal{A}^k(\mathcal{T}_{\Theta^{k-1}}(I^{k-1})), k = 1 \cdots K \end{cases}$$

where $\mathcal{A}^k$ is activate function and $\mathcal{T}_{\Theta^{k-1}}$ is affine transform $\mathcal{T}_{\Theta^{k-1}}(I^{k-1}) = \mathcal{W}^{k-1}I^{k-1} + b^{k-1}$, in which $\mathcal{W}^{k-1}, b^{k-1}$ are parameters. The parameter $\Theta = \{\Theta^{k-1} = (\mathcal{W}^{k-1}, b^{k-1}) | k = 0, \cdots K-1\}$. The output $I^K \in \{0,1\}^{C\times D\times H\times W}$ have C channels that represent C classes.

For the activate function $\mathcal{A}^K$, suppose the output of $\mathcal{T}_{\Theta^{K-1}}$ is $o^K \in \mathcal{R}^{C\times D\times H\times W}$, we want to find a output

$U \in \mathcal{W}^{C\times D\times H\times W}$ which satisfy the following problem

$$\begin{aligned} &\min - < U, o^K > + < U, \log U > \\ &s.t. \sum_c^C U_{ckij} = 1 \\ &\forall k = 1\cdots D,\ i = 1\cdots H, j = 1\cdots W \end{aligned} \tag{1}$$

In fact, the minimizer of the above problem is softmax activation function, i.e.

$$U^* = softmax(o^K),$$

where

$$U_c^* = \frac{\exp(o_c^K)}{\sum_c \exp(o_c^K)}, \quad c = 1\cdots C.$$

In this problem, we consider the regularized softmax:

$$\begin{aligned} &\min - < U, o^K > + < U, \log U > + \lambda TV(U) \\ &s.t. \sum_c^C U_{ckij} = 1 \\ &\forall k = 1\cdots D,\ i = 1\cdots H, j = 1\cdots W. \end{aligned} \tag{2}$$

where TV means the mask is piece-wise constant. The above problem is equivalent to

$$\begin{aligned} (U^*, \eta^*) = \min_U \max_{V\in\mathbb{B}} &- < U, o^K > + < U, \log U > + \lambda < U, divV > \\ &s.t. \sum_c^C U_{ckij} = 1 \\ &\forall k = 1\cdots D,\ i = 1\cdots H, j = 1\cdots W \end{aligned} \tag{3}$$

where $\mathbb{B} = \{V \in C_0^1 |\ \|V\|_\infty = \max\{\|V\|_2\} \le 1\}$. The solution of the above min-max problem satisfies the following relationship:

$$U_c^* = \frac{\exp(o_c^K - \lambda\eta_c^*)}{\sum_c \exp(o_c^K - \lambda\eta_c^*)}, \quad c = 1\cdots C.$$

We use an iterative way to find the solution:

$$\begin{cases} V^{t+1} = V^t - \tau\lambda\nabla U^t \\ \eta^{t+1} = \mathcal{P}_{\mathbb{B}}(V^{t+1}) \\ U^{t+1} = \mathcal{S}(o^K - \lambda div(\eta^{t+1})), \end{cases}$$

where $\mathcal{P}_{\mathbb{B}}$ is a projection operator onto the convex set $\mathbb{B}$,

$$\mathcal{P}_{\mathbb{B}}(V) = \begin{cases} V & if \|V\|_2 \le 1, \\ \dfrac{V}{\|V\|_2} & if \|V\|_2 > 1. \end{cases}$$

In our method, we perform just one iteration and set V^0, η^0 to 0. Then the one iteration scheme could be simplified as:

$$\begin{cases} V = -\tau\lambda\nabla\mathcal{S}(o^K), \\ \eta = \mathcal{P}_{\mathbb{B}}(V), \\ U = \mathcal{S}(o^K - \lambda div(\eta)). \end{cases}$$

In our implementation, λ is also set as learnable parameter and $\tau\lambda$ can be seen as step size which is fixed.

2.2. Network Architecture

Figure 1 illustrates the applied 3D Regularized U-Net, where the U-Net [12] architecture is adopted.

2.3. Preprocessing

None

2.4. Proposed Method

- Network architecture details: we use a standard attention unet structure as [12] provided.
- Loss function: we use the summation between Dice loss and cross entropy loss because it has been proved to be robust [13] in medical image segmentation tasks.

  For the C classes segmentation, the output of the Neural Network $\mathcal{N}_\Theta(I) = U = ((\mathcal{N}_\Theta(I))_1, \cdots (\mathcal{N}_\Theta(I))_C) \in \mathcal{R}^{C\times D\times H\times W}$. The groudtruth of the mask is $M = (M_1, \cdots M_C) \in \mathcal{R}^{C\times D\times H\times W}$

$$\mathcal{L}_{Entropy}(\mathcal{N}_\Theta(I), M) = -\frac{1}{C}\sum_{c=1}^{C} < M_c, \log(\mathcal{N}_\Theta(I))_c > \tag{4}$$

$$\mathcal{L}_{Dice}(\mathcal{N}_\Theta(I), M) = \sum_{c=1}^{C}\left(1 - 2\frac{\sum M_c * (\mathcal{N}_\Theta(I))_c + \epsilon}{\sum M_c + \sum(\mathcal{N}_\Theta(I))_c + \epsilon}\right) \tag{5}$$

- Number of model parameters: 8271238.
- Number of flops: 750361509889.

2.5. Post-processing

None

3. Dataset and Evaluation Metrics

3.1. Dataset

- A short description of the dataset used:
  The dataset used of FLARE2021 is adapted from MSD [14] (Liver [15], Spleen, Pancreas), NIH Pancreas [16, 17, 18], KiTS [19, 20], and Nanjing University under the license permission. For more detail information of the dataset, please refer to the challenge website and [21].

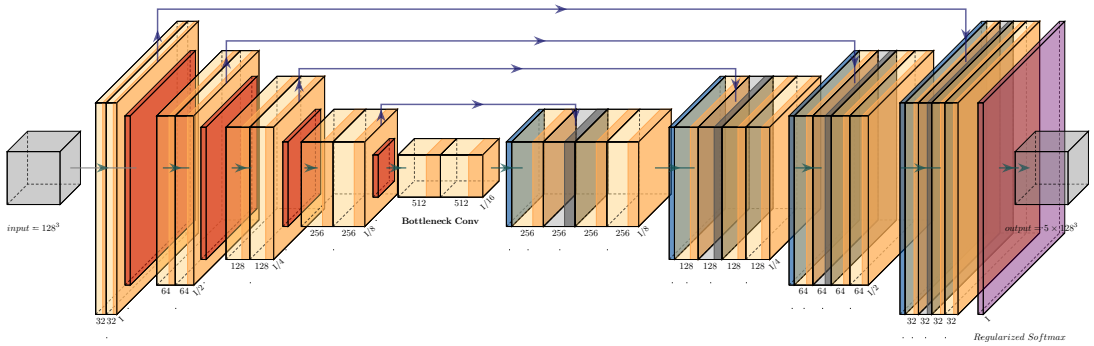


Figure 1. Network architecture

Table 1. Data splits of FLARE2021.

Data Split	Center	Phase	# Num.
Training (361 cases)	The National Institutes of Health Clinical Center	portal venous phase	80
	Memorial Sloan Kettering Cancer Center	portal venous phase	281
Validation (50 cases)	Memorial Sloan Kettering Cancer Center	portal venous phase	5
	University of Minnesota	late arterial phase	25
	7 Medical Centers	various phases	20
Testing (100 cases)	Memorial Sloan Kettering Cancer Center	portal venous phase	5
	University of Minnesota	late arterial phase	25
	7 Medical Centers	various phases	20
	Nanjing University	various phases	50

- Details of training / validation / testing splits:
  The total number of cases is 511. An approximate 70%/10%/20% train/validation/testing split is employed resulting in 361 training cases, 50 validation cases, and 100 testing cases. The detail information is presented in Table 1.

3.2. Evaluation Metrics

- Dice Similarity Coefficient (DSC)
- Normalized Surface Distance (NSD)
- Running time
- Maximum used GPU memory (when the inference is stable)

4. Implementation Details

4.1. Environments and requirements

The environments and requirements of the baseline method is shown in Table 2.

4.2. Training protocols

Full description of the training protocols, including but not limited to the items illustrated in Table 3.

The training protocols of the baseline method is shown in Table 3.

Table 2. Environments and requirements.

Windows/Ubuntu version	Ubuntu 18.04.4 LTS
CPU	Intel(R) Xeon(R) Gold 6148 CPU @ 2.40GHz
RAM	128GB
GPU	Nvidia TITAN XP
CUDA version	10.0
Programming language	Python3.7
Deep learning framework	Pytorch (Torch 1.1.0)

4.3. Testing protocols

Description of inference strategy to get the finial output on test dataset.

- Pre-processing steps of the network inputs:
  The same strategy is applied as trainging steps.

- Post-processing steps of the network outputs:
  For the class 1,3,4, we take the largest connected component as the final result. Since the human body has two kidneys, we preserve the largest three connected components to provide a result with relatively large tolerance for class 2. In many cases, the third largest connected component for class 2 is an isolated voxel

Table 3. Training protocols.

Data augmentation methods	We downsample the image to patch-size spatially. e.g. for a image I with size $C \times 512 \times 512$, the patch with size $128 \times 128 \times 128$ is genarated as $I[k : k + 128, i : 4 : 512, j : 4 : 512]$, where $k \in \{0, \cdots C - 128\}$ $i, j \in \{0, 1, 2, 3\}$ are seleted radomly.
Initialization of the network	Default
Patch sampling strategy	We select patch with size $128 \times 128 \times 128$. For the image with the patch size less than 128, we padding the image with replicate method.
Batch size	2
Input size	$128 \times 128 \times 128$
Total epochs	50 epochs
Optimizer	Adam optimizer
Initial learning rate	0.01
Learning rate decay schedule	Fixed
Stopping criteria, and optimal model selection criteria	Stopping criterion is reaching the maximum number of epoch (50).
Training time	8 hours 50 mins

and that will not cause much confusion in reality.

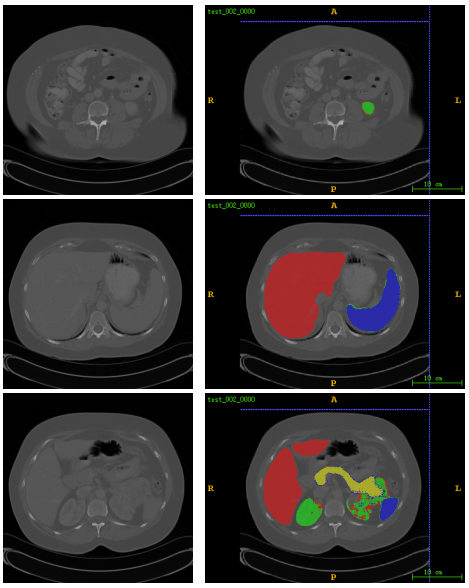
5. Results

5.1. Quantitative results on validation set.

Table 4 illustrates the results on validation cases. Figure **??** is the corresponding violin plots of the organ segmentation performance. For DSC, though the high DSC values and low dispersed distributions from the violin plots of the liver segmentation indicate great performance, the results degrade for other organs. For NSD, the obtained values and the dispersed distributions observed from the violin plots indicate unsatisfying segmentation performance for all four organs. It is worth pointing out that for liver segmentation, the DSC scores are 94.5% , indicating great segmentation performance in terms of region overlap between the ground truth and the segmented region. NSD values are 79% demonstrating that the boundary regions contain more segmentation errors, which need further improvements [21].

Table 4. Quantitative results on validation set with average value.

Organ	DSC (%)	NSD (%)
Liver	xx	xx
Kidney	xx	xx
Spleen	xx	xx
Pancreas	xx	xx



(a) Input (b) prediction

Figure 2. The input slice image, the predicted slice image and the given mask of the input image are shown in(a), (b) and (c) respectively.

Comparison between Table **??** and Table 4 illustrates better performance is obtained for the 5-fold cross validation than the validation set. This phenomenon may caused by the trained model over-fitted on training set.

5.2. Qualitative results

Figure 2 presents some challenging examples. It can be found that our method temporarily can not segment the lesion-affected organs well. But we believe our methods will make progress later.

6. Discussion and Conclusion

The result often get more than one connected components. In a few cases, we can not judge which components represents the organ. Except the Kidney organ, we take the largest connected component to present the organ. For the kidney organ, we choose the largest three components to show the result although normal people have only two kidneys.

The reason why there are multiply components is due to the training strategy that we take as input the patches of a sample data. Although the largest component post-processing operation is used, there may be betterapproaches to

Acknowledgment

The authors of this paper declare that the segmentation method they implemented for participation in the FLARE

challenge has not used any pre-trained models nor additional datasets other than those provided by the organizers.

References

[1] J. J. Cerrolaza, M. Reyes, R. M. Summers, M. Á. González-Ballester, and M. G. Linguraru, "Automatic multi-resolution shape modeling of multi-organ structures," *Medical image analysis*, vol. 25, no. 1, pp. 11–21, 2015. 1

[2] T. Okada, M. G. Linguraru, M. Hori, R. M. Summers, N. Tomiyama, and Y. Sato, "Abdominal multi-organ segmentation from ct images using conditional shape–location and unsupervised intensity priors," *Medical image analysis*, vol. 26, no. 1, pp. 1–18, 2015. 1

[3] Z. Xu, R. P. Burke, C. P. Lee, R. B. Baucom, B. K. Poulose, R. G. Abramson, and B. A. Landman, "Efficient multi-atlas abdominal segmentation on clinically acquired ct with simple context learning," *Medical image analysis*, vol. 24, no. 1, pp. 18–27, 2015. 1

[4] T. Tong, R. Wolz, Z. Wang, Q. Gao, K. Misawa, M. Fujiwara, K. Mori, J. V. Hajnal, and D. Rueckert, "Discriminative dictionary learning for abdominal multi-organ segmentation," *Medical image analysis*, vol. 23, no. 1, pp. 92–104, 2015. 1

[5] M. Suzuki, M. G. Linguraru, and K. Okada, "Multi-organ segmentation with missing organs in abdominal ct images," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2012, pp. 418–425. 1

[6] A. Shimizu, R. Ohno, T. Ikegami, H. Kobatake, S. Nawano, and D. Smutek, "Segmentation of multiple organs in non-contrast 3d abdominal ct images," *International journal of computer assisted radiology and surgery*, vol. 2, no. 3, pp. 135–142, 2007. 1

[7] P. Campadelli, E. Casiraghi, S. Pratissoli, and G. Lombardi, "Automatic abdominal organ segmentation from ct images," *ELCVIA: electronic letters on computer vision and image analysis*, pp. 1–14, 2009. 1

[8] H. Lombaert, D. Zikic, A. Criminisi, and N. Ayache, "Laplacian forests: Semantic image segmentation by guided bagging," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2014, pp. 496–504. 1

[9] B. He, C. Huang, and F. Jia, "Fully automatic multi-organ segmentation based on multi-boost learning and statistical shape model search." in *VISCERAL Challenge@ ISBI*, 2015, pp. 18–21. 1

[10] E. Gibson, F. Giganti, Y. Hu, E. Bonmati, S. Bandula, K. Gurusamy, B. Davidson, S. P. Pereira, M. J. Clarkson, and D. C. Barratt, "Automatic multi-organ segmentation on abdominal ct with dense v-networks," *IEEE transactions on medical imaging*, vol. 37, no. 8, pp. 1822–1834, 2018. 1

[11] Y. Lei, Y. Fu, T. Wang, R. L. Qiu, W. J. Curran, T. Liu, and X. Yang, "Deep learning in multi-organ segmentation," *arXiv preprint arXiv:2001.10619*, 2020. 1

[12] O. Oktay, J. Schlemper, L. L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N. Y. Hammerla, B. Kainz *et al.*, "Attention u-net: Learning where to look for the pancreas," *arXiv preprint arXiv:1804.03999*, 2018. 2

[13] J. Ma, J. Chen, M. Ng, R. Huang, Y. Li, C. Li, X. Yang, and A. L. Martel, "Loss odyssey in medical image segmentation," *Medical Image Analysis*, vol. 71, p. 102035, 2021. 2

[14] A. L. Simpson, M. Antonelli, S. Bakas, M. Bilello, K. Farahani, B. Van Ginneken, A. Kopp-Schneider, B. A. Landman, G. Litjens, B. Menze *et al.*, "A large annotated medical image dataset for the development and evaluation of segmentation algorithms," *arXiv preprint arXiv:1902.09063*, 2019. 2

[15] P. Bilic, P. F. Christ, E. Vorontsov, G. Chlebus, H. Chen, Q. Dou, C.-W. Fu, X. Han, P.-A. Heng, J. Hesser *et al.*, "The liver tumor segmentation benchmark (lits)," *arXiv preprint arXiv:1901.04056*, 2019. 2

[16] H. Roth, A. Farag, E. Turkbey, L. Lu, J. Liu, and R. Summers, "Data from pancreas-ct. the cancer imaging archive (2016)." 2

[17] H. R. Roth, L. Lu, A. Farag, H.-C. Shin, J. Liu, E. B. Turkbey, and R. M. Summers, "Deeporgan: Multi-level deep convolutional networks for automated pancreas segmentation," in *International conference on medical image computing and computer-assisted intervention*. Springer, 2015, pp. 556–564. 2

[18] K. Clark, B. Vendt, K. Smith, J. Freymann, J. Kirby, P. Koppel, S. Moore, S. Phillips, D. Maffitt, M. Pringle *et al.*, "The cancer imaging archive (tcia): maintaining and operating a public information repository," *Journal of digital imaging*, vol. 26, no. 6, pp. 1045–1057, 2013. 2

[19] N. Heller, F. Isensee, K. H. Maier-Hein, X. Hou, C. Xie, F. Li, Y. Nan, G. Mu, Z. Lin, M. Han *et al.*, "The state of the art in kidney and kidney tumor segmentation in contrast-enhanced ct imaging: Results of the kits19 challenge," *Medical Image Analysis*, vol. 67, p. 101821, 2021. 2

[20] N. Heller, S. McSweeney, M. T. Peterson, S. Peterson, J. Rickman, B. Stai, R. Tejpaul, M. Oestreich, P. Blake, J. Rosenberg *et al.*, "An international challenge to use artificial intelligence to define the state-of-the-art in kidney and kidney tumor segmentation in ct imaging." *American Society of Clinical Oncology*, vol. 38, no. 6, pp. 626–626, 2020. 2

[21] J. Ma, Y. Zhang, S. Gu, C. Zhu, C. Ge, Y. Zhang, X. An, C. Wang, Q. Wang, X. Liu, S. Cao, Q. Zhang, S. Liu, Y. Wang, Y. Li, J. He, and X. Yang, "Abdomenct-1k: Is abdominal organ segmentation a solved problem?" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 2, 4