FLARE21 Report: Multi-organ Segmentation with Length Constraint

Eric K[1]

sbpt123456@126.com

Abstract. The task of segmentation is typically defined as assigning a label representing a category to each voxel of object of interest in a medical image. Medical image segmentation plays an important roles in medical image computing and computer assisted intervention system. Along with deep learning based methods have shown significant performance improvements compared with traditional methods in natural image scenario, many powerful frameworks focusing on medical image segmentation are proposed, e.g., nnU-Net [1], MONAI[1]. In this challenge, we do multi-organ segmentation via nnU-Net framework. Although nnU-Net is powerful to give outstanding segmentation results, there are also some outlier predictions. To improve this phenomenon, we introduce length constraint into our loss function referring to active contour models. [2]

Keywords: Multi-organ segmentation · nnU-Net · Length constraint

1 Method

1.1 Network

The deep learning method we choose here is *3d_lowres* version of nnU-Net [1]. It is essentially a U-Net but with specific network architecture parameters and training parameters tuned to the characteristics of the dataset itself. Together with powerful data augmentation function, nnU-Net [1] achieves better results than other deep learning methods for many medical image segmentation tasks. The details of network structural setting please refer to Sect. 3.2.

1.2 Loss function

In our experiment setting, the whole loss function $\mathcal{L}$ is composed by $\mathcal{L}_{seg}$ for learning knowledge of organs and $\mathcal{L}_{lenght}$ for supplying constraint on outlier predictions. For $\mathcal{L}_{seg}$, cross-entropy loss combined with Dice loss [3] is the most commonly used in medical image segmentation. The $\mathcal{L}_{lenght}$ is illustrate in eq. 1:

$$\mathcal{L}_{length} = \int_C |\nabla p| \, ds = \sum_{\Omega}^{i=1,j=1} \sqrt{\left| \left(\nabla p_{x_{i,j}}\right)^2 + \left(\nabla p_{y_{i,j}}\right)^2 \right| + \epsilon} \tag{1}$$

[1] https://github.com/Project-MONAI/MONAI/

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

where p refers to prediction, C refers to curve notion in active contour model [4,5], and x and y in $p_{x_{i,j}}$ and $p_{y_{i,j}}$ mean horizontal and vertical directions respectively. ϵ is a parameter to avoid mathematical error.

So the whole loss function in our method is shown as below:

$$\mathcal{L} = \mathcal{L}_{CE} + \mathcal{L}_{Dice} + \mathcal{L}_{length} \tag{2}$$

2 Dataset and Evaluation Metrics

2.1 Dataset

- A short description of the dataset used:
  The dataset used of FLARE2021 is adapted from MSD [6] (Liver [7], Spleen, Pancreas), NIH Pancreas [8,9,10], KiTS [11,12], and Nanjing University under the license permission. For more detail information of the dataset, please refer to the challenge website and [13].
- 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 Implementation Details

3.1 Environments and requirements

The environments of the method is shown in Table 2.

3.2 Training protocols

We divide the training dataset released by the FLARE21 challenge into 4:1 to train our model. The training protocols of the baseline method is shown in Table 3.

Table 2. Environments.

Windows/Ubuntu version	Ubuntu 18.04.5 LTS
CPU	Intel(R) Xeon(R) Gold 5218 CPU @ 2.30GHz
RAM	256GB
GPU	Nvidia RTX Titan
CUDA version	11.0
Programming language	Python3.7
Deep learning framework	Pytorch (Torch 1.3.1, torchvision 0.4.1)
Specification of dependencies	nnUNet

Table 3. Training protocols.

Data augmentation methods	Rotations, scaling, Gaussian noise, Gaussian blur, brightness, contrast, simulation of low resolution, gamma correction and mirroring.
Initialization of the network	"he" normal initialization
Patch sampling strategy	More than a third of the samples in a batch contain at least one randomly chosen foreground class which is the same as nn-Unet [1].
Batch size	2
Patch size	$48 \times 192 \times 192$
Total epochs	400
Optimizer	Adam (weight decay=3e-5)
Initial learning rate	3e-4
Learning rate decay schedule	ReduceLROnPlateau scheduler with patience equal to 30

4 Results

4.1 Quantitative results on validation set.

Table 4 illustrates the results on our validation cases (not 'ValidationImage' released from FLARE21). The metrics of DSC and HD95 are used in our evaluation setting. For comprehensively reviewing prediction, no post-processing is included in our experiment.

Table 4. Quantitative results on validation set.

Organ	DSC (%)	HD95
Liver	98.3±0.7	1.48±0.86
Kidney	95.9±4.6	2.69±7.89
Spleen	97.3±2.8	2.31±7.50
Pancreas	83.2±7.7	6.22±6.87

4.2 Qualitative results

Figure 1 presents one example of validation set in training phase (1/5 data in training set released from FLARE21). Figure 2 presents one example of validation set released from FLARE21 (w/o ground truth). In these Figures, we use the same window width and level (level: 0, width: 600). The method in this paper can not generalize well to different domain by now.

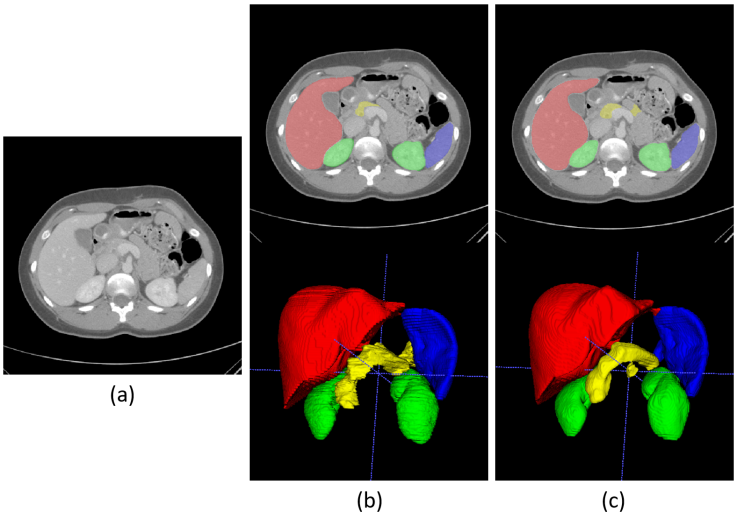

Fig. 1. Example of validation set in training phase (1/5 data in training set released from FLARE21). First column is the image, second column is the ground truth, and third column is the predicted result by our method.

Acknowledgment

The authors of this report 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.

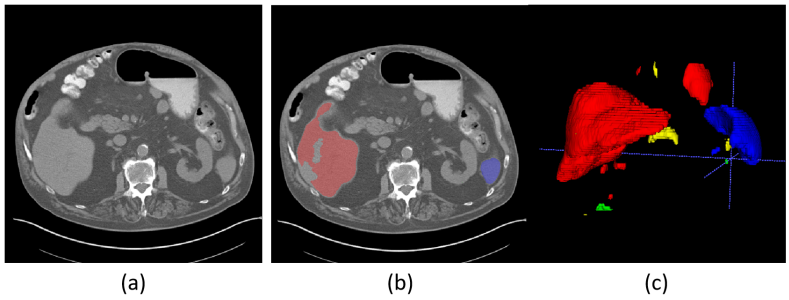

Fig. 2. Example of validation set released from FLARE21 (w/o ground truth). First column is the image, second column is the prediction, and third column is the 3D rendering of prediction.

References

1. F. Isensee, P. F. Jaeger, S. A. Kohl, J. Petersen, and K. H. Maier-Hein, "nnu-net: a self-configuring method for deep learning-based biomedical image segmentation," *Nature Methods*, vol. 18, no. 2, pp. 203–211, 2021. 1, 3
2. P. Liu, H. Han, Y. Du, H. Zhu, Y. Li, F. Gu, H. Xiao, J. Li, C. Zhao, L. Xiao *et al.*, "Deep learning to segment pelvic bones: large-scale ct datasets and baseline models," *International Journal of Computer Assisted Radiology and Surgery*, vol. 16, no. 5, pp. 749–756, 2021. 1
3. F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 fourth international conference on 3D vision (3DV)*. IEEE, 2016, pp. 565–571. 1
4. A. V. Dalca, J. Guttag, and M. R. Sabuncu, "Anatomical priors in convolutional networks for unsupervised biomedical segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9290–9299. 2
5. X. Chen, B. M. Williams, S. R. Vallabhaneni, G. Czanner, R. Williams, and Y. Zheng, "Learning active contour models for medical image segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 632–11 640. 2
6. 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
7. 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
8. H. Roth, A. Farag, E. Turkbey, L. Lu, J. Liu, and R. Summers, "Data from pancreas-ct. the cancer imaging archive (2016)." 2
9. 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
10. 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

11. 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
12. 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
13. 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