nnUNet with Signed Distance Map

Haoyu Wang, Jingyang Zhang
Shanghai Jiao Tong University
Shanghai, China
{small_dark, J.Y.Zhang}@sjtu.edu.cn

Abstract

A variety of recent studies exploit distance map to guide medical image segmentation tasks and lead to significant improvement. Motivated by this idea, we introduce the signed distance map (SDM) into nnUNet to get better performance. According to the project SegWithDistMap [1], we adopt a multi-head nnUNet to generate both probability map and distance map. We use L2 loss to regularize the regression task of the distance map. Limited by the compute capacity of the device, we trained our model for only 100 epoch in total. After conducting 5-fold cross validation experiments, our new model achieves mean dice score of 90.24% of all classes.

[We will write other parts of this paper in further versions.]

1. Introduction

Introduce the difficulties of the segmentation task and your ideas.

2. Method

A detail description of the method used, a schematic representation of the method is recommended.

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

2.1. Preprocessing

Full description of any pre-processing strategy, how the data is cleaned.

The baseline method includes the following preprocessing steps:

- Cropping strategy: None.

[1] https://github.com/JunMa11/SegWithDistMap

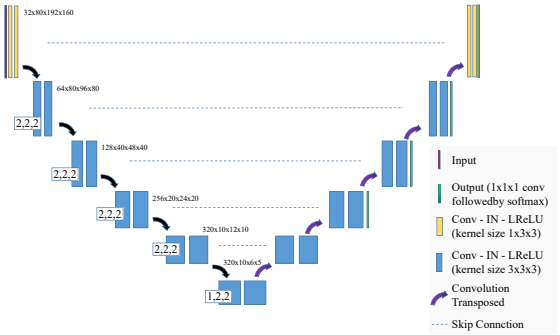

Figure 1. Network architecture

- Resampling method for anisotropic data:
  In-plane with third-order spline interpolation, out-of-plane with nearest neighbor interpolation.

- Intensity normalization method:
  First, the dataset is clipped to the [0.5, 99.5] percentiles of the intensity values of the training dataset. Then a z-score normalization is applied based on the mean and standard deviation of the intensity values.

2.2. Proposed Method

Full description of the proposed method. ***Pre-trained models are not allowed to use in this challenge.***

- Network architecture details: detail of each layer, hyper-parameters, such as strides, weights size, etc. If a standard network is used, indicate the modification:

  3D nnU-Net [2, 1] is used as shown in Figure 1, all the hyper-parameters are set as the defaulted ones.

- Loss function: we use the summation between Dice loss and cross entropy loss because it has been proved to be robust [3] in medical image segmentation tasks.

- Number of model parameters: 41268192 (can be computed via such as torchsummary library for Pytorch)

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

- Number of flops: 590861472000 (can be computed via such as fvcore library for Pytorch)

2.3. Post-processing

Description of post-processing of the model outputs to get the final output in training stage.

A connected component analysis of all ground truth labels is applied on training data [2].

3. Dataset and Evaluation Metrics

3.1. Dataset

- A short description of the dataset used:
  The dataset used of FLARE2021 is adapted from MSD [4] (Liver [5], Spleen, Pancreas), NIH Pancreas [6, 7, 8], KiTS [9, 10], and Nanjing University under the license permission. For more detail information of the dataset, please refer to the challenge website and [11].

- 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)

Table 2. Environments and requirements.

Windows/Ubuntu version	Ubuntu 18.04.5 LTS
CPU	Intel(R) Core(TM) i9-7900X CPU@3.30GHz
RAM	16×4GB; 2.67MT/s
GPU	Nvidia V100
CUDA version	11.0
Programming language	Python3.6
Deep learning framework	Pytorch (Torch 1.1.0, torchvision 0.2.2)
Specification of dependencies	nnUNet
(Optional) code is publicly available at	FLARE21nnUNetBaseline

4. Implementation Details

4.1. Environments and requirements

A description of the environment used for deployment of the method, including but not limited to the items illustrated in Table 2.

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.

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:
  No post-processing step is used.

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 [2].
Batch size	2
Patch size	80×192×160
Total epochs	1000
Optimizer	Stochastic gradient descent with nesterov momentum ($\mu = 0.99$)
Initial learning rate	0.01
Learning rate decay schedule	poly learning rate policy: $(1 - epoch/1000)^{0.9}$
Stopping criteria, and optimal model selection criteria	Stopping criterion is reaching the maximum number of epoch (1000).
Training time	72.5 hours
CO_2eq[2]	

- If using patch-based strategy, describing the patch aggregation method:
  The same patch-based strategy is applied as nnU-Net [2]. Voxels close to the center are weighted higher than those close to the border, when aggregating predictions across patches.

5. Results

5.1. Quantitative results for 5-fold cross validation.

The provided results analysis is based on the 5-fold cross validation results and validation cases.

Table 4 illustrates the results of 5-fold cross validation. Figure 2 is the corresponding violin plots of the organ segmentation performance. While high DSC and NSD scores are obtained for liver, kidney and spleen, DSC and NSD scores for pancreas indicating unsatisfactory performance. The violin plot shown in Figure 2 confirms this funding.

5.2. Quantitative results on validation set.

Table 5 illustrates the results on validation cases. Figure 3 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

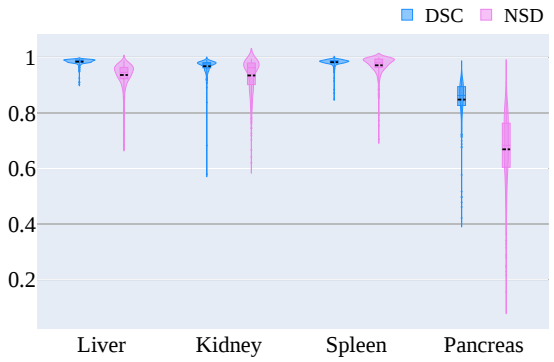

Figure 2. Violin plots of the organ segmentation results (DSC and NSD) of the 5-fold cross validation.

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 [11].

Comparison between Table 4 and Table 5 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.

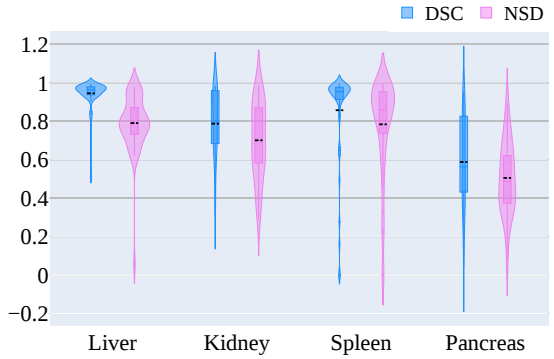

Figure 3. Violin plots of the organ segmentation results (DSC and NSD) on validation set.

5.3. Qualitative results

Figure 4 presents some challenging examples. It can be found that the baseline method can not segment the lesion-affected organs well. The first row of Figure 4 illustrates a fatty liver case where the liver is darker than healthy ones. The baseline method fails to segment the spleen (blue) and the liver (red) in this case. Second row of Figure 4 shows an example with kidney (green) tumor which causes incorrect segmentation.

6. Discussion and Conclusion

What kind of cases the proposed method works well?

Table 4. Quantitative results of 5-fold cross validation in terms of DSC and NSD.

Training	Liver		Kidney		Spleen		Pancreas	
	DSC (%)	NSD (%)	DSC (%)	NSD (%)	DSC (%)	NSD (%)	DSC (%)	NSD (%)
Fold-0	98.6±0.6	94.0±3.6	96.7±2.5	93.0±7.4	98.3±1.4	97.1±3.9	84.5±7.5	65.9±14.4
Fold-1	98.4±1.5	93.6±5.2	97.3±1.6	94.1±6.0	98.3±1.5	97.2±4.2	86.3±4.5	68.2±12.8
Fold-2	98.6±0.8	93.7±4.3	96.7±4.8	93.5±6.1	98.4±0.7	97.5±2.4	84.3±7.8	65.5±12.7
Fold-3	98.5±0.8	93.6±4.0	96.5±3.9	93.2±6.9	98.3±1.1	97.1±3.2	84.0±8.7	66.4±13.7
Fold-4	98.4±1.2	93.5±4.4	97.1±1.7	93.8±5.7	98.1±1.9	97.0±3.9	85.0±6.9	68.5±11.0
Average	98.5±1.1	93.7±4.3	96.8±3.1	93.5±6.4	98.3±1.4	97.2±3.6	84.8±7.2	66.9±13.0

Table 5. Quantitative results on validation set.

Organ	DSC (%)	NSD (%)
Liver	94.5±7.35	79.0±14.9
Kidney	78.7±17.8	70.0±19.2
Spleen	85.7±24.7	78.3±25.1
Pancreas	58.7±24.0	50.5±18.3

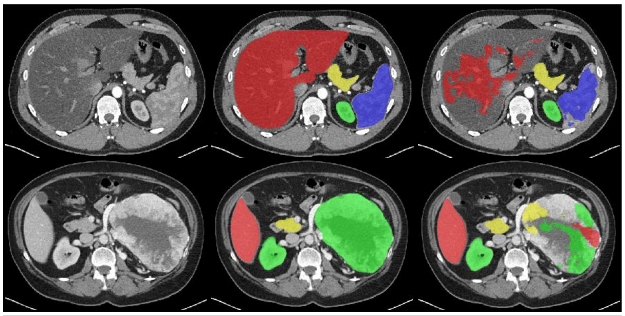

Figure 4. Challenging examples.First column is the image, second column is the ground truth, and third column is the predicted results by our baseline method [11].

The baseline method can work well on cases where no diseases exist. Besides, the DSC and NSD scores of liver segmentation is higher than the other organs, indicating liver maybe a comparable easier task as a result of its bigger size and consistent shape. Disappointing performance is obtained for pancreas segmentation as a result of the inter-patient anatomical variability of volume and shape.

What are the possible reasons for the failed cases?

The existence of leison is a critical factor for the segmentation performance. How to properly segment those cases is important. Besides, obtaining an accurate boundary segmentation need further investigate. Moreover, disappointing performance is obtained for pancreas segmentation as a result of the inter-patient anatomical variability of volume and shape.

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] O. Ronneberger, P. Fischer, and 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. 1

[2] 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, 2, 3

[3] 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. 1

[4] 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

[5] 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

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

[7] 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

[8] 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

[9] 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

[10] 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

[11] 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, 3, 4