Short Paper for MICCAI FLARE21 Challenges

Xinyun Qiu
Shenzhen University
Shenzhen, Guangdong Province, China
2070276035@email.szu.edu.cn

Abstract

We try to use a 2D semantic segmentation network to do this challenge. CT images are segmented slice by slice. We prefer the 2D model to 3D since it has faster speed on inference and lower GPU memory. Furthermore, the 2D model shows good performance almost equal to the performance of the 3D model.

1. Introduction

The main challenge is the domain adaption since training dataset and testing dataset comes from different centers. The validation dataset shares half of the same centers with testing dataset. We use a simple semi-supervised learning (SSL) framework of semantic segmentation, leveraging the unlabeled validation dataset for domain adaption.

The framework consists of a teacher model and a student model. The teacher model is trained with the training dataset and aims to provide the pseudo labels of validation dataset. The student model is trained with the training dataset and validation dataset. It aims to predict results of the testing dataset fast and efficiently. To provide higher quality pseudo labels, we use the ensemble strategy and adopt a larger network for teacher model since the inference speed and efficiency are not important.

2. Method

Both teacher model and student model are encoder-decoder structures. For the encoder head, We use Swin Transformer [12] as the backbone. The teacher model uses Swin-L while the student model use Swin-B. Both of them adopt the Feature Pyramid Networks [10] (FPN) for the decoder head. Figure 1 illustrates the adopted architecture. and Figure 2 shows the overall SSL framework.

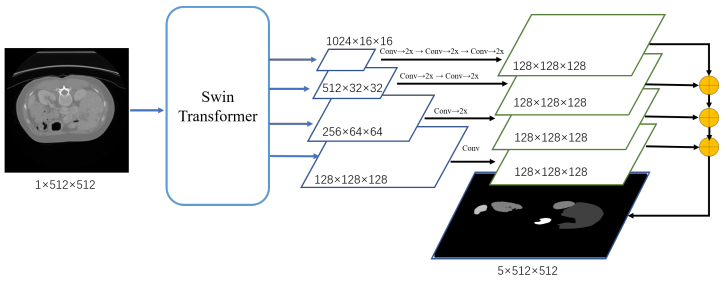

Figure 1. Network architecture

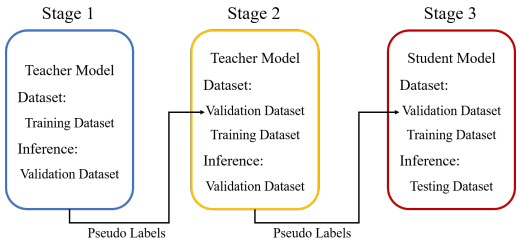

Figure 2. SSL framework

2.1. Preprocessing

The datasets are clipped to the 99.5 percentiles of the max intensity values while the min intensity values are set to -1024. Then, intensity values are normalized to the [0, 1]. For 2D segmentation, We sample 50% slices from each volume and 90% of labels are non-zero areas.

2.2. Proposed Method

The method is basically an encoder-decoder structure.

- Backbone: Swin-L (Teacher), Swin-B (Student)
- Decoder head: FPN.
- Loss function: Cross entropy.
- model parameters: 121 M (Teacher), 91 M (Student)
- flops: 296 GFLOPs (Teacher), 106 GFLOPs (Student)

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

2.3. Post-processing

To ensure the quality of pseudo labels, we set an indicator to filter out those pseudo labels with low certainty. The indicator functions is calculated as:

$$I_k = \frac{\sum_i S_i C_i}{\sum_i C_i} \tag{1}$$

where S_i and C_i donate the segmentation logit and pseudo label of the i-th pixel. C_i is a binary matrix and k represents class k . If I is higher, it means that the pseudo label is more deterministic. We set the threshold as 0.85. Pseudo labels whose I below the threshold are filtered out.

3. Dataset and Evaluation Metrics

3.1. Dataset

- A short description of the dataset used:
  The dataset used of FLARE2021 is adapted from MSD [1] (Liver [5], Spleen, Pancreas), NIH Pancreas [7, 8, 9], KiTS [6, 2], and Nanjing University under the license permission. For more detail information of the dataset, please refer to the challenge website and [4].

- 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.

Table 2. Environments and requirements.

Windows/Ubuntu version	Ubuntu 18.04.5 LTS
CPU	Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20GHz
RAM	32GB; 2.67MT/s
GPU	Titan XP
CUDA version	11.1
Programming language	Python3.8
Deep learning framework	Pytorch (Torch 1.8.1, torchvision 0.9.1)
dependencies	mmseg

4.2. Training protocols

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

Table 3. Training protocols.

Data augmentation	Flip
Initialization of the network	"he" normal initialization
Batch size	8
Image size	512×512×1
Total iterations	80000
Optimizer	AdamW
Initial learning rate	0.00006
Learning rate decay schedule	poly learning rate policy: $(1 - epoch/1000)^{0.9}$
Training time	28 hours

4.3. Testing protocols

- Pre-processing steps of the network inputs:
  The same strategy is applied as trainging steps.
- Post-processing steps of the network outputs:
  To obtain fast inference speed, there is no post-processing step.
- All slices extracted from each volume are segmented.

5. Results

5.1. Quantitative results on validation set.

Table 4 shows some comparsions between nnUNet [13] baseline and our teacher model on validation cases. On the whole, ours is slightly better than the baseline for DSC. Furthermore, The preprocessing method used in the previous validation is different from that used currently. The score of DSC and NSD could be higher. For the running time, our method has a great advantage. For GPU memory, the two are very close.

Table 4. Quantitative results on validation set.

Metric	nnUNet Baseline (%)	Ours (Teacher) (%)
Liver-DSC	94.5±8.09	93.2±14.3
Liver-NSD	79.3±14.9	77.5±16.4
Kidney-DSC	80.4±17.0	85.1±15.2
Kidney-NSD	70.9±18.4	72.0±19.4
Spleen-DSC	89.5±18.0	92.8±13.5
Spleen-NSD	82.0±19.3	85.6±18.1
Pancreas-DSC	60.1±23.1	65.1±20.3
Pancreas-NSD	50.6±17.7	49.3±17.9
Runing Time	145	54.3
GPU Memory	2298	2092

5.2. Qualitative results

Figure 3 presents some examples of qualitative results. Row 1 to row 4 illustrates that our methods can obtain better performance than baseline in some cases with lesions-affected especially the kidneys. Row 5 to row 6 shows the baseline method also fails to segment some normal organs. The difference of preprocessing method may contribute some influence.

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] 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

[2] 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

[3] "NIH Pancreas," https://wiki.cancerimagingarchive.net/display/Public/Pancreas-CT, 2020, [Online; Accessed: Aug. 2020].

[4] J. Ma, Y. Zhang, S. Gu, Y. Zhang, C. Zhu, Q. Wang, X. Liu, X. An, C. Ge, S. Cao *et al.*, "Abdomenct-1k: Is abdominal organ segmentation a solved problem?" *arXiv preprint arXiv:2010.14808*, 2020. 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] 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

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

[8] 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

[9] 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

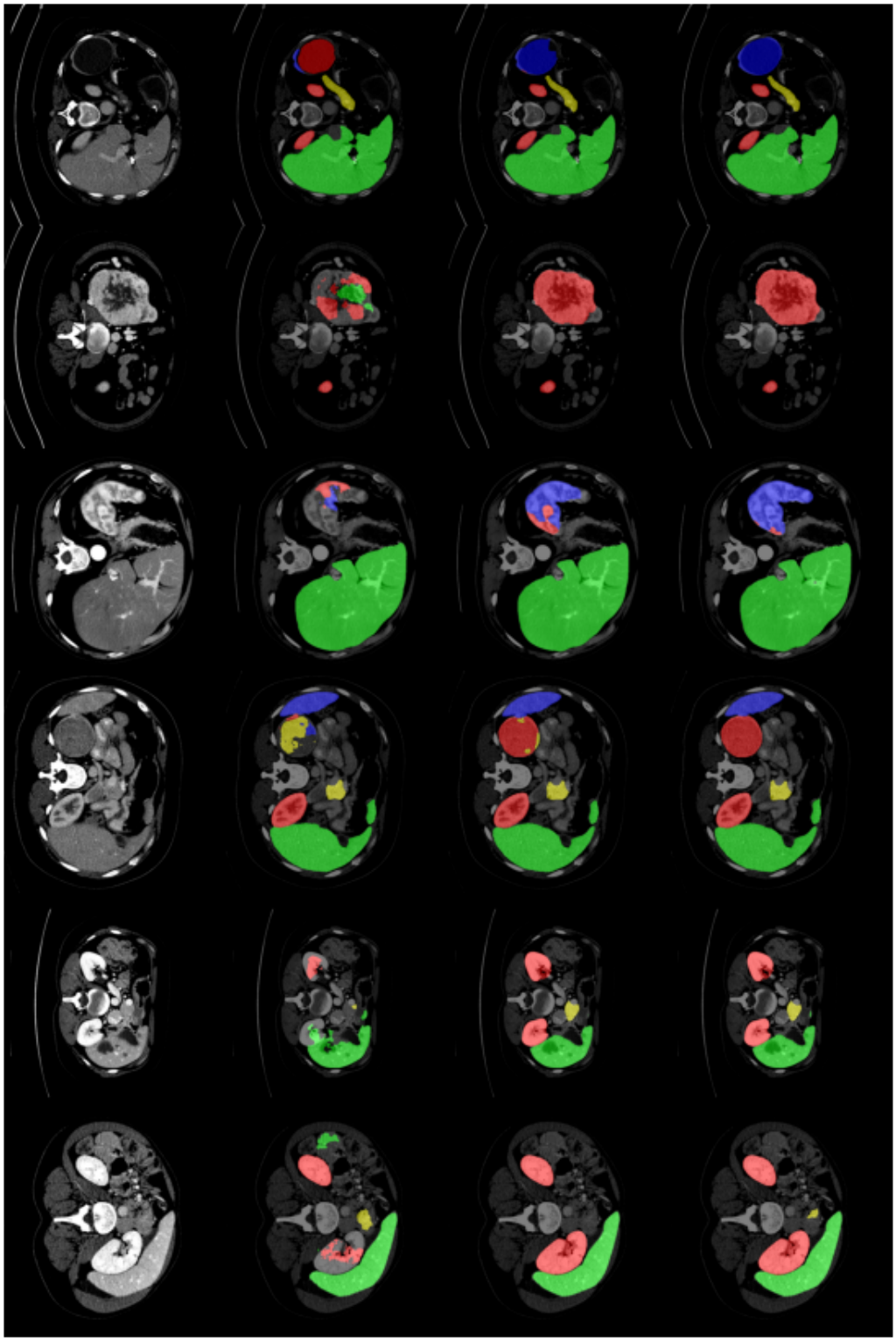

Figure 3. From left to right are origin images, baseline results, teacher model results and student model result. Red color represents kidney, blue denotes spleen, green indicates liver and yellow is the pancreas.

[10] A. Kirillov, R. Girshick, K. He, and P. Dollár, "Panoptic feature pyramid networks," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 6392–6401. 1

[11] M. Contributors, "MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark," https://github.com/open-mmlab/mmsegmentation, 2020.

[12] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," *arXiv preprint arXiv:2103.14030*, 2021. 1

[13] 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. 3