nnU-Net for Automated Lesion Segmentation in Whole-body FDG-PET/CT

Jun Ma[1,2] and Bo Wang[1,2,3,4]

[1] Department of Laboratory Medicine and Pathobiology, University of Toronto, Toronto, Canada
[2] Peter Munk Cardiac Centre, University Health Network, Toronto, Canada
[3] Department of Computer Science, University of Toronto, Toronto, Canada
[4] Vector Institute, Toronto, Canada

Abstract. Positron Emission Tomography / Computed Tomography (PET/CT) has wide applications in the diagnostic workup for various malignant solid tumor entities. In this paper, we employ the 3D full resolution nnU-Net for automatic lesion segmentation in whole-body PET/CT scans. Compared to the default nnUNet, we make three main modifications: using more data augmentations and the DiceTopK loss function and increasing the number of epochs to 1200. The final solution is an ensemble of 13 models without testing time augmentation. The ensemble model ranks top seven on the final testing set. The code is publicly available at https://github.com/JunMa11/PETCTSeg.

Keywords: Tumor Segmentation · PET/CT · U-Net.

1 Introduction

Segmentation is one of the most popular tasks in medical image analysis, aiming to generate the contour of organs and lesions of interest. During the past four years, nnU-Net [3] has been the widely used tool for medical image segmentation, which consistently achieve state-of-the-art results on various segmentation tasks. In this paper, we apply the 3D full resolution nnU-Net for automatic lesion segmentation in FDG-PET and CT scans.

There are four main challenges for lesion segmentation in PET/CT images:

- identifying the lesion needs both PET and CT images. The segmentation model should explore the complementary information between PET and CT images.
- the lesion size is relative small compared to the whole-body scans. Thus, this is a highly class-imbalanced segmentation task.
- part of the testing images are from new medical sites. The segmentation model should generalize to unseen images.
- some normal tissues (e.g., brain, bladder) have similar appearances to lesions in PET. The segmentation model should be robust to these false positives.

2 Method

To address the above challenges, we apply several modifications to the default settings of nnU-Net [3], aiming to maximize the segmentation performance.

2.1 Preprocessing

The default preprocessing in nnU-Net was used. Specifically,

- Resampling: all the images are resampled to the same target spacing: (3, 2, 2). PET and CT images are resampled with third-order spline interpolation while masks are resampled with nearest-neighbor interpolation.
- PET images are normalized by z-scoring. CT images are normalized with 0.5 and 99.5 percentiles of the foreground voxels for clipping as well as the global foreground mean and standard deviation.
- Cropping is not used in this task because the CT and PET images have few zero background regions.

2.2 Network Architecture

The network architecture is a naive 3D U-Net where the topology (e.g., the number of convolution kernels and pooling operators) is automatically configured by nnU-Net. Figure 1 illustrates the applied 3D nnU-Net.

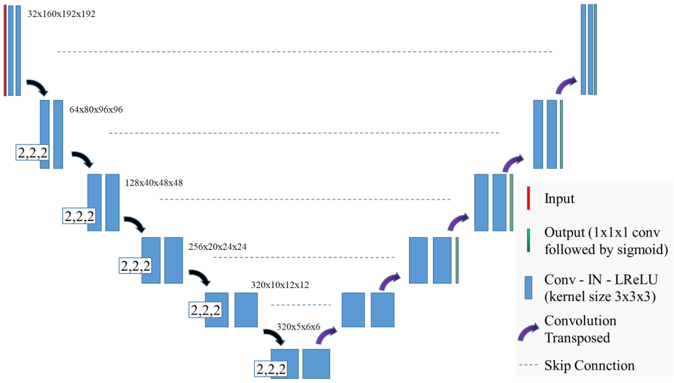


Fig. 1. 3D U-Net architecture. Paired PET and CT are concatenated as a two-channel input with the shape 2 × 160 × 192 × 192.

The PET and CT images are concatenated as a joint input, allowing the network to automatically extract complementary features for lesion segmentation. The default loss function is the summation between Dice loss and cross entropy loss. In order to alleviate the class imbalance, we also used the summation between Dice loss and TopK loss because compound loss functions have been proved to be robust in various medical image segmentation tasks [4].

To improve the generalization ability on unseen images, more data augmentations are used during training, including mirroring, rotation, scaling, Gaussian blurring, median filtering, brightness, Gamma correction, and sharpening. The trainer is available at `https://github.com/MIC-DKFZ/nnUNet/blob/master/nnunet/training/network_training/nnUNet_variants/data_augmentation/nnUNetTrainerV2_DA5.py`.

Moreover, there is resource limitations of the final model. In particular, the maximum running time is 20 minutes per cases and the RAM consumption should be within 30GB. We disable the testing-time augmentation during inference to reduce the resource consumption.

2.3 Post-processing

To reduce potential label false positives, we apply two heuristic rules as post-processing:

- If the number of foreground voxels in a segmentation mask is less than 10, these voxels will be removed from the mask.
- For each lesion, if the mean CT intensities is less than -1000 or the standard deviation is less than 5, the lesion will be removed from the mask.

3 Experiments

3.1 Dataset and evaluation measures

We only used the official autoPET datasets during model development [2][1]. There are 1014 paired PET/CT images in total. We remove the cases with no lesions or tiny lesions. Finally, 454 cases were used to train the model via five-fold cross-validation. We did not have an internal testing set and the official hidden testing set was used for final testing. The evaluation measures consist of foreground Dice score and volume of false positive and false negative.

3.2 Training protocols

The development environments and requirements are presented in Table 1.

Table 2 shows the detailed training settings.

4 Results and discussion

4.1 Quantitative analysis on cross-validation results

Table 3 shows the five-fold cross-validation results of three different method settings: the default nnUNet, using more data augmentations and changing to

Table 1. Development environments and requirements.

System	Centos
CPU	AMD Milan 7413 @ 2.65 GHz 128M cache L3
RAM	60GB
GPU (number and type)	One NVIDIA A100 40G
CUDA version	11.6
Programming language	Python 3.9
Deep learning framework	Pytorch (Torch 1.10, torchvision 0.2.2)
Specific dependencies	nnU-Net
Code	`https://github.com/JunMa11/PETCTSeg`

Table 2. Training setting.

Network initialization	“he" normal initialization
Batch size	2
Patch size	160×192×192
Total epochs	1200
Optimizer	SGD with nesterov momentum ($\mu = 0.99$)
Initial learning rate (lr)	0.01
Lr decay schedule	$(1 - \frac{epoch}{epoch_{max}})^{0.9}$
Loss function	Dice+CE and Dice+TopK10 [4]
Trainable parameters	31,195,648

the loss function to the summation between Dice loss and TopK loss. It can be found that none of the method can consistently achieve the best performance across different folds and metrics. Thus, we opt to use ensembles as the final model. The DA5 model in fold 2 and the DA5-TopK10 model are not used because of their inferior performance. The final model is the ensemble of 13 models.

4.2 Qualitative analysis of false positives and false negatives

Figure 2 and Figure 3 show some typical examples of false positives and false negatives, respectively. The false positives usually have high intensities on PET images while the false negatives have relatively low intensities on PET images.

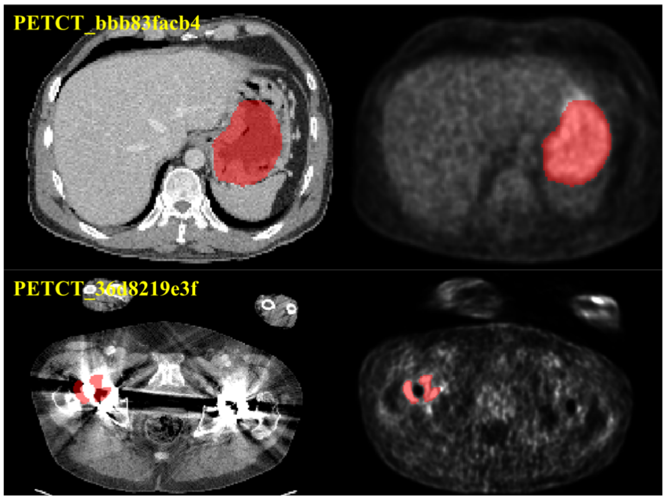
4.3 Failed attempt

We also tried to train a two-class classifier to classify the segmented lesions as true positive and false positive. Specifically, we extracted radiomics features with pyradiomics [5] and trained an AutoGluon[5] model. However, the classification accuracy was 0.7, which is not satisfied. Thus, this strategy was not used in the final solution.

[5] https://auto.gluon.ai/stable/index.html

Table 3. Quantitative results based on cross-validation. Bold number denotes best results in each group. The DA5 model in fold 2 and the DA5-TopK10 model are not used. The final ensemble model is based on remaining 13 models.

Fold	Methods	Dice Score	False Positive Volume	False Negative Volume
0	Default	**0.7717**	8.9465	9.9940
	DA5	0.7669	10.5258	**7.6886**
	DA5-TopK10	0.7708	**7.4553**	11.0655
1	Default	**0.7561**	8.8201	4.7033
	DA5	0.7519	10.2265	**2.5304**
	DA5-TopK10	0.7664	**5.8786**	3.0903
2	Default	**0.7265**	**5.4426**	16.4931
	DA5*	0.6851	15.5450	17.4535
	DA5-TopK10	0.7214	7.9747	**16.4329**
3	Default	**0.7424**	**4.8882**	7.5890
	DA5	0.7413	6.6658	8.9288
	DA5-TopK10	0.7431	6.9327	**6.6734**
4	Default	**0.7400**	7.5384	5.5701
	DA5	0.7202	**6.6717**	8.4219
	DA5-TopK10*	0.7231	19.9575	**2.8952**



(a) CT Image (b) PET Image

Fig. 2. Typical examples of false positives. Red masks denote segmentation results.

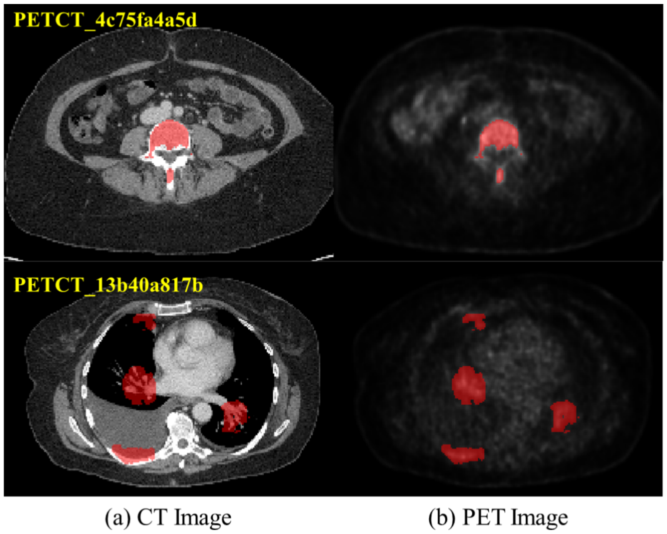


Fig. 3. Typical examples of false negatives. Red masks denote ground truth. These tumors are missed in segmentation results.

5 Conclusion

In this technical report paper, we present the solution to automatic lesion segmentation from PET/CT images. The default nnU-Net is modified with more training epochs and data augmentations. We also use different loss functions to address the label-imbalance. In addition, we identify the typical patterns of false positives and false negatives, which could be eliminated by incorporating domain knowledge.

References

1. Clark, K., Vendt, B., Smith, K., Freymann, J., Kirby, J., Koppel, P., Moore, S., Phillips, S., Maffitt, D., Pringle, M., et al.: The cancer imaging archive (tcia): maintaining and operating a public information repository. Journal of Digital Imaging **26**(6), 1045–1057 (2013) 3
2. Gatidis, S., Küstner, T., Ingrisch, M., Fabritius, M., Cyran, C.: Automated Lesion Segmentation in Whole-Body FDG- PET/CT (Mar 2022). https://doi.org/10.5281/zenodo.6362493, https://doi.org/10.5281/zenodo.6362493 3
3. Isensee, F., Jaeger, P.F., Kohl, S.A., Petersen, J., Maier-Hein, K.H.: nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. Nature Methods **18**(2), 203–211 (2021) 1, 2
4. Ma, J., Chen, J., Ng, M., Huang, R., Li, Y., Li, C., Yang, X., Martel, A.L.: Loss odyssey in medical image segmentation. Medical Image Analysis **71**, 102035 (2021) 3, 4

5. Van Griethuysen, J.J., Fedorov, A., Parmar, C., Hosny, A., Aucoin, N., Narayan, V., Beets-Tan, R.G., Fillion-Robin, J.C., Pieper, S., Aerts, H.J.: Computational radiomics system to decode the radiographic phenotype. Cancer research **77**(21), e104–e107 (2017) 4