Report of KiTS19 competition

Du Bang[1,2,*] , Yu Hongyun[1,2,*], Zheng Xiangshang[1,2,*],Wang Wenzhe[1,2,*], Ying Haochao[1,2,*,#]

[1] College of Computer Science and Technology, Zhejiang University
[2] Real Doctor AI Research Centre, Zhejiang University

1 Introduction

As an important organ of the urinary system, the kidney focuses on generating urine, purifying the blood, and also maintaining water, electrolytes and acid-base balance. Kidney tumors, as one of the most common tumors, are extremely harmful. Once they are found, surgery is the widespread treatment. Therefore, accurate segmentation of renal tumors is of great significance to surgeons performing renal tumor resection. To this end, this paper proposes an effective method for segmentation of the kidney and its tumor for the KiTS19 competition. Specifically, our method first designs a 3D ResUNet framework to segment the whole kidney, and then develops a 2.5D segmentation network to segment the tumors based on the result of kidney segmentation. After validation, the performance of our method reaches a good level under the given metric.

2 Kidney segmentation

2.1 Model Architecture

Considering that the spatial position of the kidney in CT is relatively stable and the spatial information is of great significance for the accurate localization and segmentation of the kidney, we choose 3D UNet[2] as the skeleton structure. To be specific, our network consists of a 4-layer encoder and a 4-layer decoder. On this basis, we introduce the idea of the residual module that inputting the output of the previous layer to the next layer in both the encoder and decoder after the necessary sampling of the output. In addition, in order to make the information of each decoder more efficient, we refer to the deep supervision idea and design one map layer following each decoder layer. That is, the map layer transforms each output of the decoder to the output of the target shape, which regards as part of the output of the entire network. Finally, our network outputs a total of 4 target-matrix, and the loss calculation and back propagation are performed together for the 4 matrices.

We choose Dice Loss as our loss function, defined as follows:

$$L = \mu(L_1 + L_2 + L_3) + L4$$

* equal contribution
# corresponding author. haochaoying@zju.edu.cn

Where L_i represents Dice Loss between each output matrix and annotation. The initial value of μ is 0.33, which are reduced to 80% of the previous one after per 15 epochs of training.

2.2 Data

The data set we used contains 210 labeled training data and 90 unlabeled test data provided by the event. The training data set mainly consists of delay period CTs, the slice size is 512 × 512, the layer thickness is between 0.75 and 5 mm, and the average layer thickness is about 3 mm.

In order to unify the data input and try to keep enough input information, we use the method of trilinear interpolation to unify the layer thickness of all training data to 2mm. Further, we use the nearest neighbor interpolation method to transform corresponding segmentation mask so as to keep the shape of CT and annotation consistent. Due to the limitation of computing resource, the original slice size is not suitable as an input, so we reduce each slice size to 256 × 256.

Since the volume of the kidney is small compared to other organs, many slices in one case do not contain the kidneys. Thus, in order to balance the positive and negative samples, we refer to the annotation data and select out the slices with kidney annotation to expand up and down 50 times. In addition, considering that the amount of data in the training set is relatively small, and one person's two kidneys are roughly symmetrical in the body, we perform a centrally symmetric mirroring operation in each case data to augment the data set.

Finally, to train our model, we randomly choose 190 of annotated data as training sets and the rest as validation sets.

2.3 Training

We conduct all our experiments on two TITAN Xp GPUs using the Pytorch 1.1 deep learning platform. During training, we randomly selected 64 consecutive slices from each case into the network to ensure maximum spatial information. The training process ends at 500 epoches, which takes approximately 25 hours.

2.4 Validation

In the verification phase, we took the idea of sliding the window. The slice plane is the XOY plane, and the slice stacking direction is the Z axis. The general idea is as follows:

```
start_slice = 0
end_slice = 64
output_list = []
While end_slice <= z:
  Choose [start_slice:end_slice] slices;
  Forward;
```

```
  Output_list.append(output_array);
  start_slice = start_slice + 32;
  end_slice = start_slice + 64 – 1;
for output_array in output_list:
  out_label = concatenate(out_label,
             output_array[16:48])
loss_function(out_label, annotation_label);
```

3 Renal tumors segmentation

3.1 Data preparation

After the kidney segmentation in above section, the mask image of kidney can be obtained. To further segment the tumors in the kidney, we multiply the mask image and original image to reduce the possible influence of background and also guide our model to focus on the kidney region. After that, trilinear interpolation operation, as in section 2, is used to unify the slice thickness in each case. In this part, the thickness is unified to 1 mm for keeping more Z-axis information. Then, we normalize the HU value (40-50) in kidney region in each slice[6]. In addition, we also observe that the proportion of the slices containing the kidneys is not high. Therefore, we remove the slices without the kidneys to avoid our model to learn a lot of unnecessary information.

Finally, the original size 512 × 512 in each slice is used as the input of our model. Moreover, due to the disorder property of the data set itself, we simply choose the No. 0-194 cases as the training set and the rest as the validation set. Otherwise, to obtain more training data, the image and corresponding label are randomly rotated, translated, cut, histogram averaging and Gaussian blurring to increase the amount of data.

3.2 Model Architecture

To effectively segment the tumors, we try various deep learning model for segmentation, including UNet, VNet, DeepLab V3+. In the experiment, we find that VNet outperforms UNet and DeepLab V3+ in the segmentation of small targets[1][2]. We also observe the performance of 3D VNet is not as well as that of 2.5D VNet. Therefore, we choose the 2.5D version of VNet as our final model framework. In addition, we design the following mechanisms to further improve the segmentation performance of tumors.

Multi-scale learning. In the encoder part, each down-sampling operation has an additional branch input operation, which is to down-sample the original image to different scales and the output image size of each down-sampling operation is the same. By comparing the segmentation results of multi-scale VNet with the original VNet, it can be found that multi-scale is better in feature learning and the segmentation effect has been significantly improved, which shows that multi-scale VNet is useful.

SENet module. To distinguish the different contributions of feature channels, we add SENet module into our multi-scale VNet. Specifically, a SENet module is added to the down-sampling of each layer to filter the features of the layer[3]. After adding SENet module, the segmentation effect of multi-scale VNet has been improved to some extent, which proves that SE module is useful.

Attention mechanism. In medical images, salient features (such as related tissues or organs) are useful for specific tasks, which suppresses irrelevant areas in input images. Therefore, to enforce our model to focus on the part of kidney tumors, we refer to attention gate in Attention-UNet[4]. That is, in each short connection of multi-scale VNet, the features extracted by encoder and the corresponding features of decoder are concatenated together to make an attention gate, which is the whole part as a new short connection. The advantage of this operation will make the features from corresponding down-sampling and up-sampling more targeted. After training this model, the segmentation results can be improved greatly, which proves that the addition of attention module is meaningful.

Dilated convolution. Dilated convolution is well understood as adding reception fields by injecting holes into standard convolution maps. However, there is also a problem with dilated convolution, that is, the Gridding Effect will appear in the individual expansion convolution. In addition, long-range information may not be relevant is also a problem, because using large dilation rate to obtain information may only be effective for segmentation of some large objects, but may be harmful for small objects. Therefore, it is necessary to select the appropriate dilation rate and location to use. With regard to the Gridding Effect, we used HDC, Hybrid Dilated Convolution, in the competition to avoid this problem[5]. To prevent the emergence of the Gridding Effect, the expansion convolutions of successive dilated rates are superimposed to fill each other's voids. The second problem, that is, long-range information may be not relevant, is also very simple, trying to use HDC at each level, comparing the results of the model can determine which layer is the best to use HDC. After trying, we use three layers of HDC at the bottom of encode, dilated rate is, 2, 3, 5. After adding HDC, the segmentation effect is improved, which shows that the dilated convolution is useful.

3.3 Loss Function

The original loss function is dice loss, which is the evaluation index of this competition. However, the result is not good in training. After observing the data, we find that the data of the validation set is very unbalanced, and there are many false positive samples. After trying to use the loss function which combines focal loss and weighted dice loss, the performance are improved.

3.4 Resample

After experimenting with various models and adding several modules, we observe that the dice score of the model on the validation set is always low and steadily oscillates around 0.45. As a result, this performance cannot be accepted. Fortunately, we find that about 40% of the samples had renal cysts, that is to say, the probability of false positive samples was higher. Because renal cysts were similar to renal tumors and the HU value is almost same, the model might regard renal cysts as renal tumors. In order to avoid identifying false positive samples, we need to learn more and more about these difficult samples, so we argue that resampling the difficult samples to train our model will be helpful. After resampling, the dice score segmentation of the validation set has been significantly improved, reaching about 0.7. This shows that resampling is very useful in the face of difficult samples.

3.5 Testing

The operation of the test set is similar to that of the previous training set and the data preprocessing of the verification set, including the linear interpolation of slice thickness, HU value normalization, etc. The only difference is that the label of kidney is obtained through our kidney segmentation model. After preprocessing the data, the segmentation of renal tumors is carried out. After generating the segmentation data, we perform a posteriori operation on the data. Because the result of segmentation may not be accurate, it may be segmented into false positive data, such as kidney cysts. In order to avoid the impact of these data on real prediction, we use the maximum connectivity component to optimize the generated data.

4 Reference

1. Milletari, F., Navab, N., & Ahmadi, S. A. (2016, October). V-net: Fully convolutional neural networks for volumetric medical image segmentation. In 2016 Fourth International Conference on 3D Vision (3DV) (pp. 565-571). IEEE.
2. Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (pp. 234-241). Springer, Cham.
3. Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7132-7141).
4. Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M., Heinrich, M., Misawa, K., ... & Glocker, B. (2018). Attention u-net: Learning where to look for the pancreas. arXiv preprint arXiv:1804.03999.
5. Wang, P., Chen, P., Yuan, Y., Liu, D., Huang, Z., Hou, X., & Cottrell, G. (2018, March). Understanding convolution for semantic segmentation. In 2018 IEEE winter conference on applications of computer vision (WACV) (pp. 1451-1460). IEEE.
6. Cuingnet, R., Prevost, R., Lesage, D., Cohen, L. D., Mory, B., & Ardon, R. (2012, October). Automatic detection and segmentation of kidneys in 3D CT images using random forests. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 66-74). Springer, Berlin, Heidelberg.