Kidney Tumor Segmentation Using 3D U-nets for Masking and Labeling

Yuichi Ito[1], Takuya Nishimoto[1], Weizhong Chou[2], Toshitaka Ishiguro[2],
Kensaku Mori[2], Kosuke Kojo[2], Takahiro Kojima[2],
Tsutomu Nakagawa[1], and Hideki Kakeya[2]

[1] DNP, 1-1-1 Ichigaya-Kagacho, Shinjuku-ku, Tokyo 162-8001, JAPAN
[2] University of Tsukuba, 1-1-1 Tennodai, Tsukuba 305-8577, JAPAN
Itou-Y46@mail.dnp.co.jp

Abstract. In this paper we report on the algorithm we applied to KiTS 19 Challenge. We used 3D U-nets for masking and labeling separately. We mainly focused on data augmentation and grouping of training data to improve the result of segmentation.

Keywords: 3D U-net, training data, data augmentation, data grouping

1 Introduction

Since the idea of deep convolutional neural networks (CNNs) was proposed [1], they have been widely applied to medical image processing. Since the proposal of U-net [2], which is based on fully convolutional network (FCN) [3], deep CNNs have been applied to various biomedical image segmentation tasks and have outperformed the conventional algorithms. To apply deep CNNs to 3D volume data, 3D U-net has been proposed [4], where three dimensional convolutions are applied to attain volumetric segmentation. 3D U-net is easily applied to multi-organ CT segmentation or tumor segmentation, both of which are important processes for computer-aided diagnosis and therapy.

To tackle the segmentation task given by KiTS 19 Challenge, we apply two different 3D U-nets for masking and labeling respectively. To improve the segmentation result, we also apply data augmentation and grouping to balance the ratio of training data to feed to the 3D U-net. We explain the masking step in Section 2 and the labeling step in Section 3.

2 Masking Step

In the first step of our algorithm, we make a mask that occludes the part other than kidneys and kidney tumors. To attain this goal, we apply the following neural networks.

The masking step is composed of two processes. In the first process we apply a large scale 3D U-net for raw CT data. (Here and hereafter the raw data is the interpolated data provided by the contest web site.) Because of the limitation of memory usage for 3D convolution, the largest patch size we can train is around 128 x 128 x 64 with our computer, which installs the latest NVIDIA machine learning flagship graphics card GV100 with 32 GB video memory. Here we apply padding. The structure of the 3D U-net we used is shown in Fig. 1. The patch size we used was 128 x 128 x 32, which allowed us to increase the batch size to 8.

In our machine learning we applied histogram equalization to preprocess CT images. To avoid extreme equalization, we use a parameter α to adjust the degree of histogram averaging. We make a probabilistic distribution given by

$$\alpha P_a(x) + (1 - \alpha)\, P_b(x), \tag{1}$$

where P_a represents the probabilistic distribution of the completely uniform distribution and P_b is the probabilistic histogram of the CT voxel HU values. We used $\alpha = 0.25$ and applied histogram equalization to the probabilistic distribution synthesized above.

Also we introduce a batch weighted loss function for training. The weight is set so that it becomes larger when the number of voxels included in each category decreases. To avoid excessive weighting, we used a weight in inverse proportion to the cube root of voxel numbers.

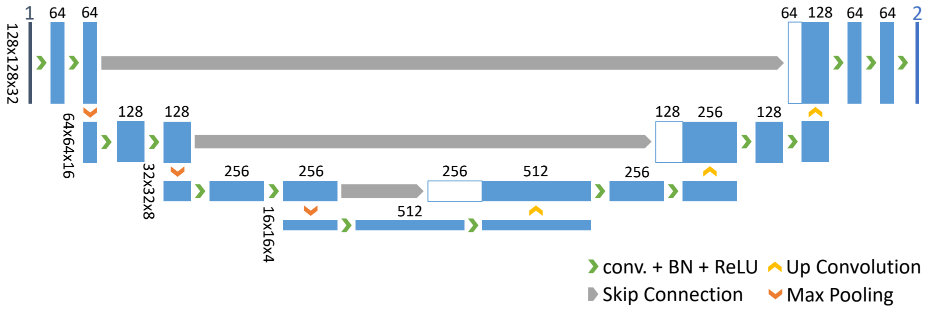

Fig. 1. Structure of 3D U-net for masking.

Though the above 3D U-net can make a mask that reflects the detailed shape of kidneys, speckles and debris emerge in the result. To remove the noises, first we apply a cleanup program using the ITK Toolkit. To be concrete, we remove speckles and debris that are composed of less than 50,000 voxels. With this process, small noises are removed with ease.

In the second process, we use another 3D U-net to monitor the overall features of the abdominal domain to remove large noises. We first down-sample the size of CT data from 512 x 512 to 256 x 256 in XY dimension. Then we feed the down-sampled data to the 3D U-net with the same structure as shown in Fig. 1. Because the patch size of the neural network is large and the resolution of the original data is reduced, the neural network can read a considerable range of features at a time, which greatly reduces the possibility of error marking in a completely unrelated domain.

When testing the neural network, we first down-sample the original CT data and then feed them to the neural network as blocks. After the segmentation results are obtained, we up-sample the label data and restore the 512 x 512 label data in XY dimension. After removing speckles and debris composed of less than 50,000 voxels here again, we choose the two areas where the heights of the centers are close to the average height of kidneys. (Note that one area is chosen when only one area remains after removing noises.)

Then we pick up the areas where the mask given by the first process and that given by the second process overlap with each other. We apply OR operation between the overlapping areas given by the first and second processes to make the final mask for labeling.

3 Labeling Step

Based on the masking result in the first step, the unmasked parts are divided into three classes: kidney, tumors, and the others. Here again we apply 3D U-net for segmentation. The patch size we used was 48 x 48 x 16. We apply padding here also. The structure of the 3D U-net we used is shown in Fig. 2. Here we do not apply histogram equalization nor weighted loss used in the masking step.

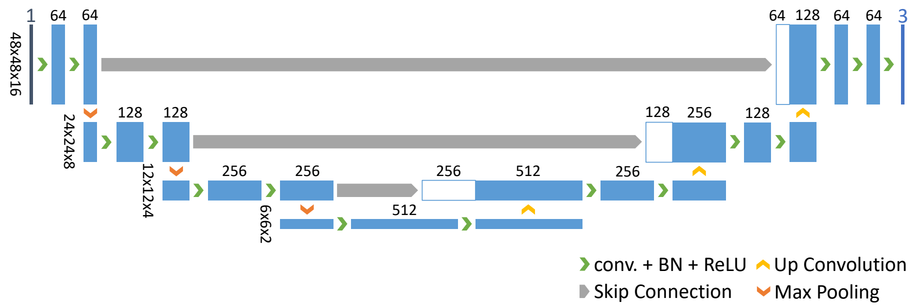

Fig. 2. Structure of 3D U-net for segmentation.

The feature of our system for segmentation is preparation of training data set. First we categorized the types of tumors. After simple observation, according to whether the size of cancer exceeds the size of the kidney, and according to whether the overall color of the kidney is darker than the shade, we roughly divided the tumors into the following four categories:

Small dark tumors,
Large dark tumors,
Small light tumors,
Large light tumors.

This classification can be easily calculated automatically using dimension statistics and average HU calculations using scripts.

In the training process 80% of data are used for training, while 20% of data are used for validation. To balance the ratio of tumors in each category during training, the data are divided so that 80% of tumors in each category may be used for training and the others may be used for validation. In addition, the abnormal data (#005, #015, #037, #151) such as horseshoe kidney and labelling errors are removed from training data.

Preparation of patches in our training process has the following two features. First we removed the patches that do not include tumor parts from the training data set to focus on tumors in the training process.

Second we apply data augmentation to enhance training. The data are randomly contracted or expanded by the factor of 0.8-1.2, from which the data are randomly cropped to feed to the network. The stride is 12 x 12 x 4 and only the central 24 x 24 x 12 part of the output layer is used as the result.

Post-processing to remove small noises comprises the following three processes. First we apply AND operation between the obtained label and the dilated mask. Second the speckles smaller than 100 voxels are removed. Third the tumors dissociated from the kidney are removed.

4 Conclusion

To solve the segmentation problem given by KiTS 19 Challenge, we applied 3D U-nets for masking and labeling respectively. For effective training, we introduced data augmentation and grouping to balance the ratio of training data, which improved the results of segmentation.

References

1. Krizhevsky, A., Sutskever, I., Hinton, G. E.: Imagenet classification with deep convolutional neural networks, Advances in Neural Information Processing Systems 1, pp. 1097–1105 (2012).
2. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Heidelberg (2015).
3. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation, arXiv:1411.4038 (2014).
4. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. In: Ourselin, S., Joskowicz, L., Sabuncu, M., Unal, G., Wells, W. (eds) MICCAI 2016, LNCS, vol. 9901, pp. 424–432. Springer, Cham (2016)