Using Two-stage Network to Segment Kidneys and Kidney Tumors

Pan Chen, Chenghai Xu, Jie He, Chengwei Sun,
Yingying Ma, and Fenglong Sun

Digital China Health Technologies Co., Ltd.

Abstract. There are many new cases of kidney cancer each year, and surgery is the most common treatment. To assist doctors in surgical planning, an accurate and automatic kidney and tumor segmentation method is helpful in the clinical practice. In this paper, we propose a deep learning framework for the segmentation of kidneys and tumors in abdominal CT images. The key idea is using a two-stage strategy. First, for each case, we use a 3d U-shape convolution network to get the localization of each kidney. Then using next 3d U-shape convolution network we obtain the precise segmentation results of each kidney. Finally, merge the results to obtain the complete segmentation. Also, we try some tricks to improve the performance.

Keywords: kidney tumor segmentation, deep learning, two-stage

1 Introduction

There are more than 400,000 new cases of kidney cancer each year, and surgery is its most common treatment. Due to the wide variety in kidney and kidney tumor morphology, there is currently great interest in how tumor morphology relates to surgical outcomes, as well as in developing advanced surgical planning techniques. Automatic semantic segmentation is a promising tool for these efforts, but morphological heterogeneity makes it a difficult problem. In this paper, inspired by [1], we introduce a two-stage segmentation framework based on DCNN, consisting of 1) a localization model to detect the interest of region containing kidneys; 2) a segmentation model to focus on the region of one kidney and obtain the segmentation result. Since the CT image is three dimensional volume data and these organs are intrinsically 3d objects, DCNN filters learning directly on the overall 3d CT volume enables capturing the complete spatial context of organs. But due to the computational intensity and limit amount of GPU memory, the input image cant be very large. Our method has two advantages: on the one hand, we dont need high resolution image for localization task; on the other hand, the segmentation model can only concentrate on the important local region. Both reduce the sizes of input images and make the algorithm efficient.

2 Method

As mentioned above, we first determine the region where kidneys are located. Then we focus on this region to get fine segmentation.

2.1 Localization

Data preprocessing With CT intensity values being not standardized, normalization is critical to allow for data from different processing types. First we clip the CT at [-95, 155], and then we normalize each image independently by subtracting the mean and dividing by the standard deviation.

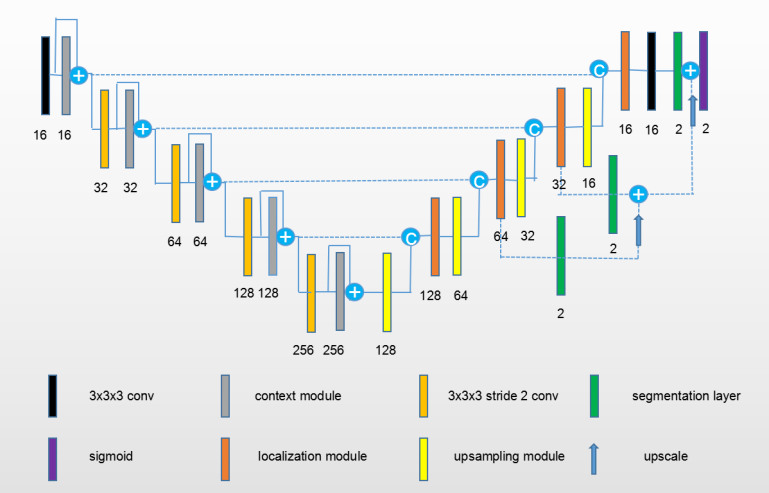

Fig. 1. Network architecture. The number below the box refers to the number of output channels

Network architecture Our network is inspired by the U-Net architecture [2] and we refer to [3]. See Fig.1. Our architecture comprises of a context pathway and a localization pathway. In the context pathway, residual blocks that we call context modules are used. Every context module consists of two 3x3x3 convolutional layers and a dropout layer ($p = 0.3$) is in between. Context modules are connected by stride 2 $3\times3\times3$ convolutions. As going deeper, the context modules give more abstract representations of lower image resolution. In the localization pathway, we have three localization modules. Every localization module consists of a $3 \times 3 \times 3$ convolution, which halves the number of channels, followed by a $1 \times 1 \times 1$ convolution. The upsampling module consists of an upsampling (size 2, stride 2) and a $3 \times 3 \times 3$ convolution which halves the number of channels.

Through upsampling and concatenating, a merged feature map is send to the localization module. We employ deep supervision in the localization pathway by integrating segmentation layers ($3 \times 3 \times 3$ convolution) at different levels of the network and combining them via element-wise summation to form the final network output. Throughout the network we use leaky ReLU non-linearities for all feature maps. We furthermore replace the traditional batch normalization before activation functions with instance normalization in the case of small batch size.

Training procedure Our network architecture is trained with $128 \times 128 \times 128$ voxels obtained by zooming the original CT images and batch size 1, due to the memory limitation of GPU. The employed optimizer is Adam with an initial learning rate $lr = 5 \times 10^{-4}$, the following learning rate schedule: after every epoch, the learning rate is reduced to be 98.5% of the original and a L2 weight decay of 10^{-5} . In order to deal with the class imbalance in the data, we use a two-class Dice loss function, i.e. the weighted average of Dice loss functions of the two classes, background and foreground (kidney and tumor) with weight 0.01 and 0.99. During training, flipping at three axes and rotating axis slices on the fly is applied to prevent overfitting. Note that we segment the image to obtain the localization.

Post-processing At inference phase, the prediction is upsampled by a nearest neighbor interpolation to match the shape of the original input scan. After that, for each organ, we remove the connect regions whose volumes are smaller than 5% of the volume of the maximal connect region.

2.2 Segmentation

Training data In essence, we use the similar network architecture and similar training procedure to obtain the segmentation model except for using different training data. Noting that the training data of the localization model is zoomed from the original CT image, so its resolution is lower, which is enough for localization, but not good for segmentation. For saving the resolution as much as possible, we clip the original data by extending the bounding box which exactly contains the groundtruth of each kidney with 10 pixels in all directions. If one side of this clipped cube is smaller than 64, we replace it by 64. That is, using a bounding box of shape $64 \times 64 \times 64$ to be the lower bound. After that, resize the cube to $128 \times 128 \times 128$ before sending it to the network.

Training procedure We only consider the difference with localization. First, to reduce the clipping error, we add the online random translation to enlarge the training data. Secondly, we use a multi-class weighted Dice loss of three classes, background, kidney and tumor, with weights 0.01, 0.14, 0.85. Of course, now the output channel number of the network should be 3, while in localization the number is 2.

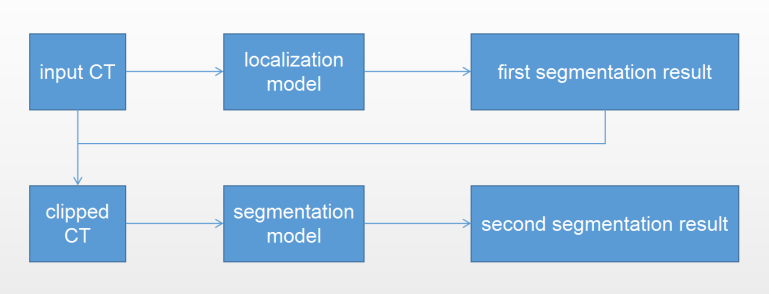

Fig. 2. two-stage segmentation

Inference procedure At inference phase of our two-stage segmentation, we first use the localization model to get a rough localization of kidneys, and basing on this information to clip the original CT image and resize the clipped cube containing only one kidney to $128 \times 128 \times 128$. Then send this resized cube to the segmentation model to obtain the fine segmentation result, which becomes the final result after post-processing and returning to the original image. See Fig.2.

3 Results

We trained and evaluated our network on the KITS19 training dataset (210 CTs) and test dataset (90 CTs)[4]. No external data was used and the network was trained from scratch. We split the training dataset into two subsets, 200 CTs for training, and 10 CTs for validation. On training subset, the kidney dice is 0.95, and the tumor dice is 0.8; on validation subset, the kidney dice is 0.8, and the tumor dice is 0.92.

References

1. Roger Trullo, et al.: Fully automated esophagus segmentation with a hierarchical deep learning approach. In:2017 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)IEEE, 2017.
2. Ronneberger, O., Fischer, P., & Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In:International Conference on Medical image computing and computer-assisted intervention, Springer, Cham, pp. 234-241, October 2015.
3. Isensee, Fabian , et al.: Brain Tumor Segmentation and Radiomics Survival Prediction: Contribution to the BRATS 2017 Challenge. In:International MICCAI Brainlesion WorkshopSpringer, Cham, 2017.
4. Heller N , Sathianathen N , Kalapara A , et al.: The KiTS19 Challenge Data: 300 Kidney Tumor Cases with Clinical Context, CT Semantic Segmentations, and Surgical Outcomes. (2019)