

# Triplanar 3D-to-2D networks with dense connections and dilated convolutions: application to the KITS 2019 challenge

Richard McKinley<sup>1</sup>, Roland Wiest<sup>1</sup>, Hendrik von Tengg-Kobligk<sup>2</sup>

<sup>1</sup>Support Centre for Advanced Neuroimaging, University Institute of Diagnostic and Interventional Neuroradiology, Inselspital, Bern University Hospital, University of Bern, Switzerland

<sup>2</sup>University Institute for Diagnostic, Interventional and Paediatric Radiology, Inselspital, Bern University Hospital, University of Bern, Switzerland

**Abstract.** We describe a method for the segmentation of kidney and kidney tumors based on computed tomography imaging, based on the KITS 2019 challenge dataset.

## Introduction

The KITS dataset [2] is a collection of CT volumes, with a large variety of slice thicknesses, in-plane resolution, and coverage, of patients with kidney tumors. The data is presented together with annotations of healthy-appearing kidney tissue and kidney tumors. We present in this short paper a description of a method to segment both kidney and tumors in this dataset, using a fully-convolutional neural network with extensive automatic post-processing.

Kidney tumors can have a variety of appearances, and can be small and completely surrounded by healthy kidney tissue, or large (larger than the remaining healthy kidney tissue) and protrude from the kidney itself. This heterogeneity, together with the variety of spatial resolutions in the KITS dataset and lack of soft tissue detail make kidney tumor segmentation a challenging task. In addition, the ratio of kidney tumor voxels to non-tumor voxels is low: only about 0.2% of voxels included in the training dataset are labelled as tumor. On a per-case basis, the ratio of non-tumor to tumor voxels ranges from 20:1 to 200000:1, with the average ratio being approximately 500:1. As a result, to train a segmentation algorithm successfully on this dataset, we will need to employ methods to ensure that the gradients coming from tumor voxels are not overwhelmed by those coming from non-tumor voxels.

The CT images in the KITS dataset are presented as axial slices, with varying slice thickness. Given this variation in the data, an obvious solution would be to train a segmentation algorithm operating on 2D slices only. However, such an algorithm would, we hypothesize, have difficulty distinguishing tumor tissue

from other healthy organs, especially where the healthy-appearing kidney tissue is not visible in the 2D slice. For this reason, we train our classifier on data which has been resampled to  $1\text{mm}^3$  isovoxels. We train a fully-convolutional neural network, with an architecture similar to one previously applied to brain-tumor segmentation, multiple sclerosis lesion segmentation, and neuroanatomy segmentation. The network is trained on anisotropic  $13 \times 180 \times 320$  sub-volumes sampled from the training data, using a 3D-to-2D architecture which predicts the labels of the centre slice of the input volume. The orientation of the anisotropic training volumes can be axial, sagittal or coronal, meaning that the resulting network can segment the target tissues in all three planes. The resulting network has extremely high performance on detecting healthy kidney tissue, but suffers from outliers in the detection of tumor tissue: we therefore first identify regions containing tissue having a high likelihood of containing kidney tissue, and mask detected tumor tissue not coming from these regions.

## Pre-processing and data handling

In order to make the problem of segmenting kidney tumors more tractable, we initially resampled all volumes to have 1mm isotropic resolution: CT images were resampled using trilinear interpolation, and label maps using nearest neighbour interpolation. Between cases, the coverage of the CT images varies dramatically: some cases show only the area immediately surrounding the kidneys, while other cases are located adjacent to various healthy or pathologic body structures, e.g. the intestine. The amount of homogenous fatty tissues varies a lot among individuals. To further standardise the inputs to our network during training, we cropped each resampled volume to five slices above the first axial slice containing kidney or tumor tissue to five slices below the last axial slices containing kidney or tumor tissue. (This causes problems in classifying tissue in regions of the body not seen during training, which we solve by applying extensive post-processing).

We windowed the CT values in the volumes, such that values below -50 were set to -50, and values above 450 were set to 450. Intensities were normalized in each training volume, by dividing by the mean of voxel intensities above -50 and below 450, and dividing by the standard deviation of those voxels.

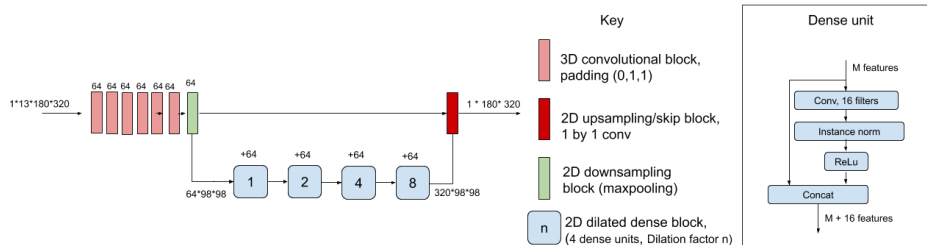
152 of the 190 available training cases were selected randomly for training the network, and 8 of the remaining cases were selected to assess performance during training.

## The modified DeepSCAN architecture

The default choice for neural network architectures has for some time been the variations on the U-net [9]. In particular, the winning architecture in the recent Medical image segmentation decathlon was based on a 3D U-net variant. [4].

However, architectures of this style have significant drawbacks: in particular, if the volume to be segmented is large, segmentation must proceed patch-by-patch. Not only does this limit the receptive field of the classifier: since segmentation accuracy deteriorates at the borders of these patches, it is often necessary to overlap the patches substantially, increasing the computation time of applying the classifier.

The DeepSCAN architecture has been recently used to segment brain tumors [6], MS lesions [8], and brain anatomy [7]. Here we apply it so kidney tumor segmentation. The architecture is pictured in Figure 1: it is broadly the same as those used in the above papers, being a hybrid of U-net with a smaller-than-usual number of pooling layers, and a bottleneck layer consisting of dense blocks [3] of dilated convolutions. In the case of kidney segmentation, instead of batch normalization we use Instance Normalization, allowing stable training on batches of size 1.



**Fig. 1.** The DeepSCAN architectures, as applied to kidney tumor segmentation

## Training

The network segments the volume slice-by slice: the input data is thirteen consecutive slices, Ground truth for such a set of slices is the lesion mask of the central slice. Input images were initially cropped to remove as much empty space as possible. Batch size during training was 1. As a result, the input tensor to the model has dimensions  $1*1*13*180*320$ . Models were trained using a cosine-annealing learning rate schedule, in which the learning rate was varied between  $1e-5$  and  $1e-9$  during each epoch.

Slices from all three directions (sagittal, axial, coronal) were fed to the classifier for training. Examples of the different segmentations in those three directions (just for the whole tumor label) can be seen in Figure ??.

The classifier was trained to segment three tissue classes independently: whole kidney (labels 1 and 2), healthy kidney (label 1), and tumor (label 2). We em-

ployed a number of techniques to combat the data imbalance in the training set between the tumor labels and the background. No gradient was taken from voxels with intensity less than or equal to -50, or voxels with intensity greater than or equal to 450. We used focal loss [5] with a gamma of 2 to concentrate on difficult-to-classify voxels. Since the segmentation of healthy kidney was observed to be a much easier task than segmenting the tumor, the loss from the tumor task was weighted 100 times more than the whole kidney and healthy kidney tasks. We heavily oversampled training examples containing the tumor class: each batch was a single  $13 \times 180 \times 320$  patch sampled from one of the training examples, and patches with tumor tissue in the central slice were preferred over slices containing no tumor, with 1 in 20 batches containing no tumor. One epoch was 1000 patches: after 200 epochs, we observed that the classifier was failing to detect any tumor tissue in several of the validation cases. Labelling of the training cases was performed by interpolation, both by the challenge organisers (in the original spacing of the KITS19 cases) and then by us (in preparing the  $1\text{mm}^3$  training data). To combat underlabeling of training examples arising from this, we ran the classifier over all training cases, and added tumor labels where the classifier predicted tumor but the ground truth showed healthy kidney. After this change, we were able to detect tumor voxels in all validation cases, and training was halted after 700 epochs.

### **Application of the classifier**

To apply our method to a new case, the imaging was resampled to  $1\text{mm}^3$  isovoxels. The classifier was applied to in axial, sagittal and coronal directions: owing to the non-isotropic input volumes, it was possible to apply the classifier without stitching patches even on cases with full-body imaging. Logits (network output before applying sigmoid) were produced for each tissue type (whole kidney, healthy kidney, and tumor) and each direction (sagittal, axial, coronal). For each tissue type, we averaged over the three directions, and then resampled the resulting maps to the resolution of the original imaging.

### **Postprocessing**

The above pipeline leads to a logit map for the whole tumor, healthy kidney, and tumor. A naive segmentation of the kidney and tumor can be obtained from the classifier as follows: label all tissue with a whole kidney logit greater than zero as kidney (label 1). Inside that kidney label, label every voxel with tumor logit greater than healthy logit as tumor (label 2).

This naive labeling suffers from the restricted training regimen: there are regions of the body not seen by the classifier during training (most often the bladder, but also distended intestines and regions of bone in the extremities) which are incorrectly labelled as tumor. As an initial post-processing step, we identify

large connected components of well-classified healthy kidney tissue (regions with healthy kidney logit greater than 0.5, with volume greater than 20000  $mm^3$ ). We then restrict our attention, initially, to connected components with kidney logit  $> 0$  which intersect with those large regions of healthy kidney. Any tissue within those regions with tumor logit  $< 0$  is labelled as kidney, and any tissue with tumor logit  $> 0$  is labelled as tumor.

Despite our extensive attempts to combat the class imbalance in the dataset, we still find that for many tumors the above method undersegments the tumor class: several tumor voxels have tumor logit  $< 0$ , especially in small tumors. In an attempt to segment the full extent of the tumor, we employ a random walk segmentation, as implemented in `scipy.ndimage` [1]: seeds for the random walk are set as tissue with tumor logit  $> 0$  in the already-identified kidney, while seeds for the background (not tumor) class are those with tumor logit  $< -2$  or healthy logit  $> 1$ . These values were determined by experimentation the validation cases. The final tumor segmentation was then given by the result of this random walk segmentation.

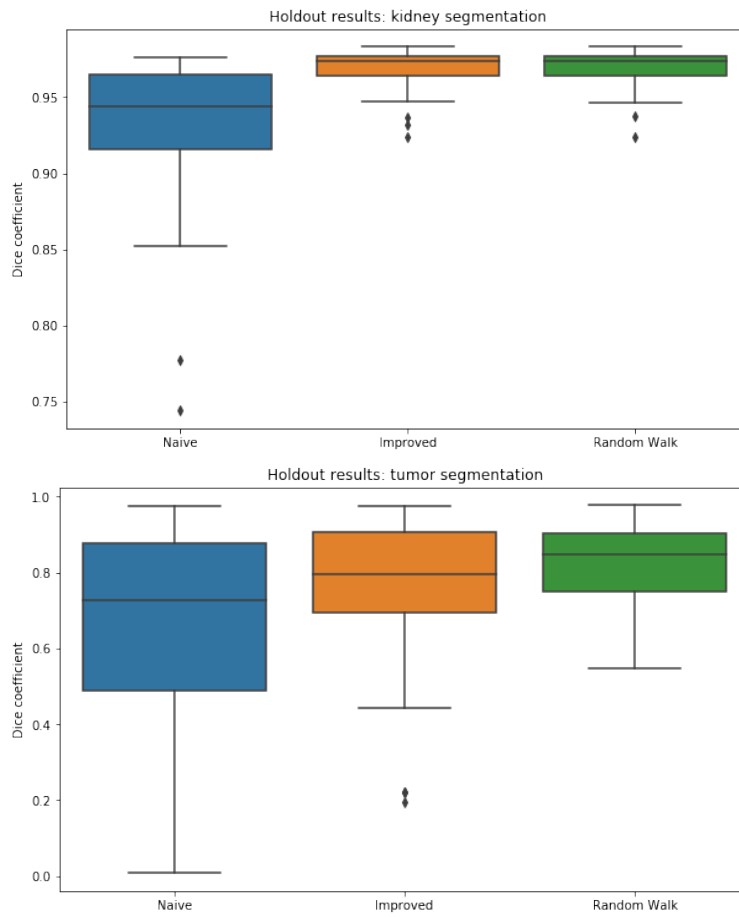
## Results

On our in-house test set, without postprocessing, our method achieved a mean Dice of 0.93 on the whole kidney (labels 1 and 2), with postprocessing but without random walk segmentation a mean dice of 0.97, and with full postprocessing including random walk segmentation a mean Dice of 0.97. On the tumor itself (label 2), without post-processing we achieve a mean Dice of 0.66, with postprocessing but without the random walk segmentation a mean Dice of 0.75, and with full post-processing a mean Dice of 0.82. Boxplots of the dice coefficients arising from our method can be seen in Figure 2.

## Discussion

## References

1. Grady, L.: Random walks for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* 28(11), 1768–1783 (Nov 2006), <https://doi.org/10.1109/TPAMI.2006.233>
2. Heller, N., Sathianathan, N.J., Kalapara, A., Walczak, E., Moore, K., Kaluzniak, H., Rosenberg, J., Blake, P., Rengel, Z., Oestreich, M., Dean, J.F., Tradewell, M., Shah, A., Tejpaul, R., Edgerton, Z., Peterson, M., Raza, S., Regmi, S., Papanikolopoulos, N., Weight, C.J.: The kits19 challenge data: 300 kidney tumor cases with clinical context, ct semantic segmentations, and surgical outcomes. *ArXiv abs/1904.00445* (2019)
3. Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017)



**Fig. 2.** Segmentation results (Dice coefficient) on the held-out training cases, as produced by the Naive algorithm (no post-processing), the Improved algorithm (with post-processing, but without random walk segmentation of the tumor), the the Random Walk algorithm

4. Isensee, F., Petersen, J., Klein, A., Zimmerer, D., Jaeger, P.F., Kohl, S., Wasserthal, J., Koehler, G., Norajitra, T., Wirkert, S.J., Maier-Hein, K.H.: nnu-net: Self-adapting framework for u-net-based medical image segmentation. ArXiv abs/1809.10486 (2018)
5. Lin, T.Y., Goyal, P., Girshick, R.B., He, K., Dollár, P.: Focal loss for dense object detection. 2017 IEEE International Conference on Computer Vision (ICCV) pp. 2999–3007 (2017)
6. McKinley, R., Meier, R., Wiest, R.: Ensembles of densely connected cnns with label-uncertainty for brain tumor segmentation. In: Crimi, A., Bakas, S., Menze, B. (eds.) Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries. Springer International Publishing, Cham (2019)
7. McKinley, R., Rebsamen, M., Meier, R., Reyes, M., Rummel, C., Wiest, R.: Few-shot brain segmentation from weakly labeled data with deep heteroscedastic multi-task networks. arXiv e-print, available at <https://arxiv.org/abs/1904.02436> (Apr 2019), <https://arxiv.org/abs/1904.02436>
8. McKinley, R., Wepfer, R., Aschwanden, F., Grunder, L., Muri, R., Rummel, C., Verma, R., Weisstanner, C., Reyes, M., Salmen, A., Chan, A., Wagner, F., Wiest, R.: Simultaneous lesion and neuroanatomy segmentation in multiple sclerosis using deep neural networks. ArXiv abs/1901.07419 (2019)
9. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation. Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015 pp. 234–241 (2015), <http://arxiv.org/abs/1505.04597> \delimitert026E30F\$nhhttp://link.springer.com/10.1007/978-3-319-24574-4\_28