KIDENY TUMOR SEGMENTATION WITH 3D-CONVOLUTIONAL NEURAL NETWORKS

Olexander Chepurnoi*
Ukrainian Catholic University
chepurny@ucu.edu.ua

Yaroslava Lochman*
Ukrainian Catholic University
lochman@ucu.edu.ua

April 2019

ABSTRACT

Will be formulated in the next iteration.

***K*eywords** Medical Volumetric Image Segmentation · 3D Convolutional Neural Network · 3D Kidney Segmentation · 3D Tumor Segmantation

1 Introduction and Related Work

This work covers a medical case of 3D image segmentation with specific convolutional neural network (CNN) applied. More precisely we solved kidney and kidney tumor segmentation problem. The project is done within KiTS19 Challenge organized by the University of Minnesota and University of Melbourne [7]. It was chosen because cancer is still one of the biggest problems in healthcare and we believe that automated medical image processing may fasten disease diagnostics and treatment. There are more than 400 000 new cases of kidney cancer each year[2]. And surgery is its most common treatment [3]. Kidney cancer is the ninth most commonly occurring cancer in men and the 14th most commonly occurring cancer in women. The top countries with the highest incidence of kidney cancer in 2018 are shown on the Figure 1. Automatic segmentation of the organ and its tumor is a promising tool. It can impact decision making processes that are now complex given the wide range of treatment options that are currently available.

Some of the earliest works (2009 year) on kidney tumor analysis first of all aimed to classify renal tumors based on their size and anatomical features and predict the risk of overall complications resulting from the nephron-sparing surgery based on the score. In [5] along with tumor size were analyzed: anterior or posterior face, longitudinal, and rim tumor location; tumor relationships with renal sinus or urinary collecting system; and percentage of tumor deepening into the kidney. An evaluating algorithm was implemented with multivariate analysis tools. In [9] the scoring system was created based on the following observed variables: maximal diameter of the tumor, exophytic or endophytic characteristics, nearness to the closest portion of the renal sinus or collecting system, location relative to the polar line and whether the tumor is anterior or posterior to the renal coronal plane.

The 3D image segmentation problem is depicted on the Figure 2. For each input case which is a volumetric data (we will call it an input volume) the goal is to get a segmentation mask volume for kidney and a segmentation mask volume for tumor.

From the medical image segmentation side there exists a very well known fully convolutional neural network called U-Net [12] that solves the general segmentation task. Based on this architecture several models such as 3D U-Net [4], Residual Symmetric U-Net [10], V-Net [11] were created to deal with volumetric images. Authors also presented data augmentation and training pipelines to handle unbalanced data, reduce false positives and overcome a limited memory budget. There also exist ICv1 [2] and Deepem3d [14] neural networks with integrated inception and residual learning

[*] Authors contributed equally

[2] Kidney Cancer Statistics 2018, www.wcrf.org/dietandcancer/cancer-trends/kidney-cancer-statistics

[3] Cancer Diagnosis and Treatment Statistics 2017, www.cancerresearchuk.org/health-professional/cancer-statistics/diagnosis-and-treatment

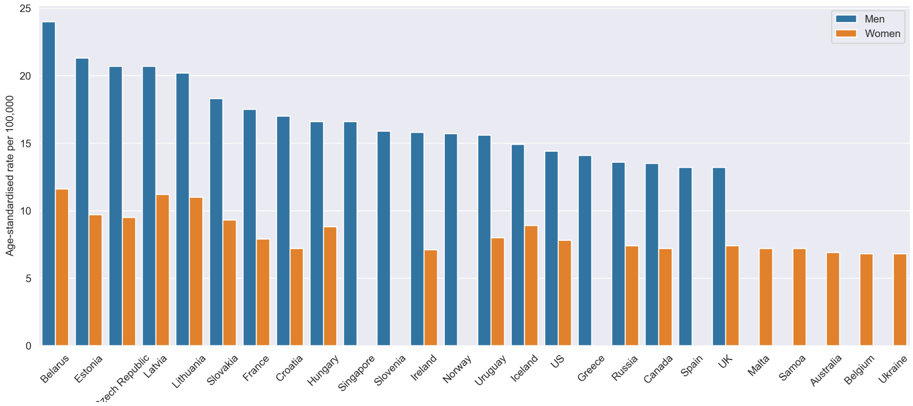


Figure 1: Kidney cancer rates in 2018. The data was drawn from www.wcrf.org/dietandcancer/cancer-trends/kidney-cancer-statistics

techniques and watershed algorithm applied to get accurate boundaries. The very close to our problem was effectively solved by Kid-Net [13] that handled aforementioned data and memory barriers as well.

In many CNN works the residual block introduced in ResNet [6] was widely used [11] [10] [13] [3] to facilitate training deep architecture. Alongside with 3D convolutions there exist approaches with recurrent plus 2D convolutional neural networks [1]. Feature accumulation with recurrent residual convolutional layers ensures better feature representation for segmentation tasks.

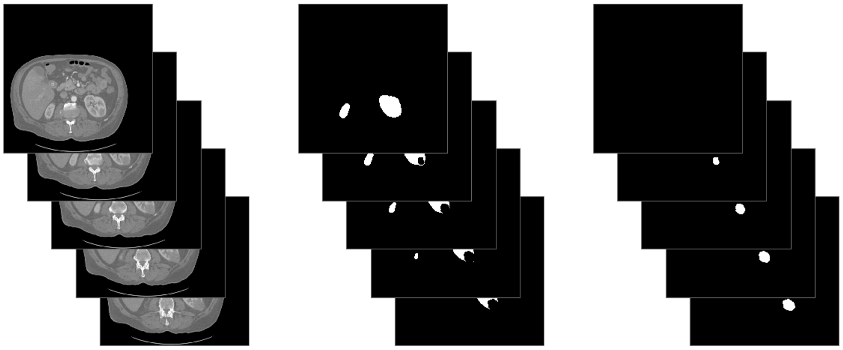


Figure 2: 3D image segmentation problem: an input is a volumetric data and can be considered as a collection of ordinary images; outputs are collections of binary masks corresponding to object classes.

Further here also our thoughts on how can the segmentation problem be transferred to other tasks will be formulated.

2 Data

The annotated dataset consists of contrast-enhanced multi-phase abdominal Computed Tomography (CT) imaging plus kidney and kidney tumor segmentation masks for 300 patients who underwent nephrectomy for kidney tumors between 2010 and 2018 [7]. The length of the training set is 210 samples, and 90 samples will be used to compute final score. The organizers of the KiTS19 contest are also planning to release the clinical attributes as a table data soon that may affect our model consideration according to that (see Section 3.1).

2.1 Exploratory Analysis

The imaging as well as ground truth labels are provided in the anonymized NIFTI format having a shape number_of_slices × height × width. Here, number_of_slices corresponds to an axial plane view, and slices go from superior

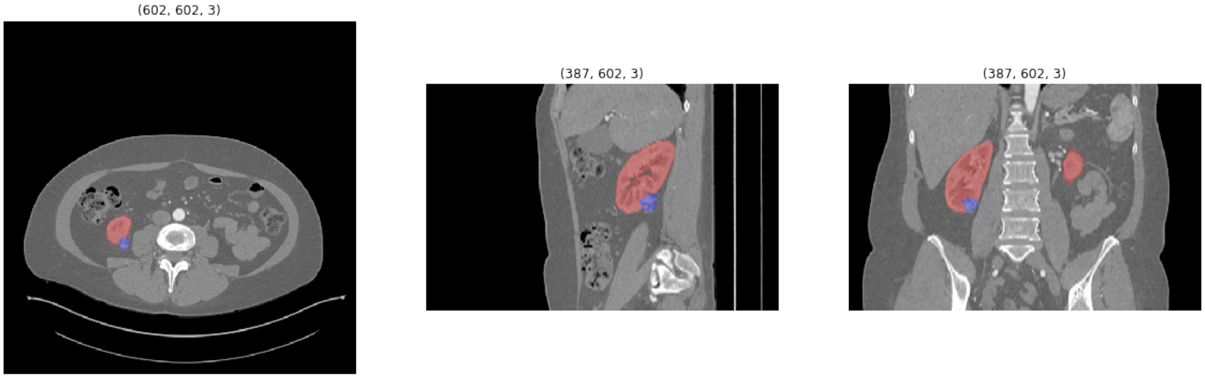
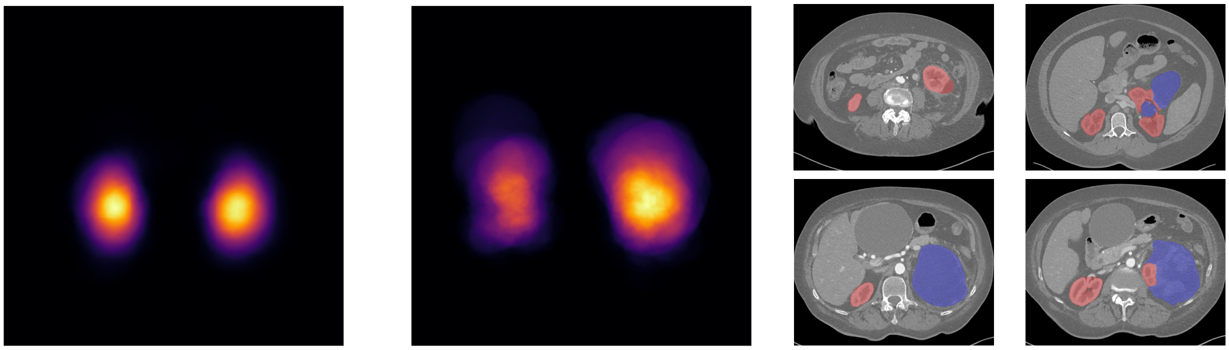


Figure 3: Training data case example: a CT scan slice through an axial (left), sagittal (middle) and coronal (right) plane. Red annotated segments correspond to a kidney, blue – to a tumor.

to inferior as the index increases. In all cases, the patient was supine during CT data collection, and thus height-width origins lie to the patient's left anterior. When there were multiple qualifying series for a particular case, those with the smaller slice thickness was chosen. The slice thicknesses range from 1 to 5 mm. For the training data example see Figure 3.

Raw Data and Interpolated Data There are two kinds of data provided. The first one was released earlier and corresponded to raw CT data. It consists of cases that for every patients have the same height and width 512×512 but different number of slices, slice thickness (the distance between captured axial sections in mm) and pixel width (the distance between centers of adjacent pixels in axial sections in mm). In fact it is unnormalized and we can't compare one case with the other. The next is data interpolated in that way so the sizes (height, width, depth) are comparable – the units are the same for all cases, proportional to the millimeters in each dimension. Our first attempt was done with the raw data but further we switched to the interpolated by above reason.

Location In every case a patient has two kidneys, one of which has a tumor, the other doesn't. As can be seen on the Figure 4 (a) there is a certain location of concentration of kidneys which helps in sampling process of crops (see Section 2.2). However the relative locations of the left kidney to the right kidney may be different meaning they are not mirrored that also affects the sampling process and data augmentation (we can't use horizontal flipping transformation). For the example see top right image on the Figure 4 (b). The tumor location is not as uniform. Considering correct data collection we may conclude that it is focused more on the right side that also implies appropriate sampling.



(a) (b)

Figure 4: (a) Mean average kidney (left) and tumor (right) location in the axial plane. (b) Edge cases: location, intersecting, sizes, boundaries.

Kidney-Tumor Boundary The kidney object and the tumor object should not overlap each other which complicates the task. It is not trivial where the kidney ends and tumor begins and there are even cases when they intersect. The Figure 4 (b) on the right illustrates some examples.

Data Balance Not only we are interested in foreground-background balance but also the ratio of kidney to tumor is important. In about 10% cases the tumors are bigger than whole kidneys (e.g. see bottom examples on the Figure 4 (b)). And as was analyzed earlier the locations should be taken into consideration.

2.2 Preprocessing

Before splitting the data we've done this:

- drop cases with bad annotation (in total 209 examples left)

The training data was split according to the distribution of the kidney size and tumor size jointly. (See Figure 5). We report results in Section 4 on the validation set given by this split. Data leakage is avoided since we deal with full cases corresponding to different patients. At the same time the validation set is as representative as it can be having data provided by the organizers.

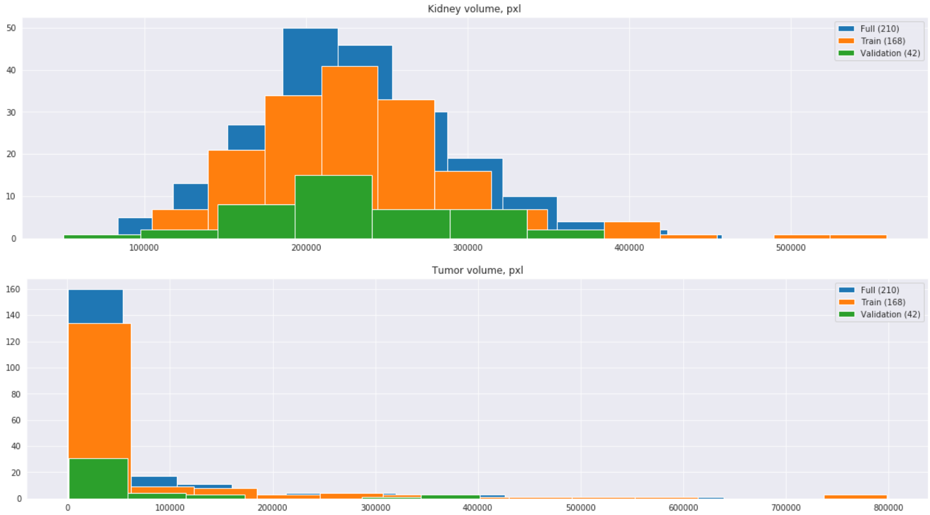


Figure 5: Distribution of kidney size (top) and tumor size (bottom) in the full 210 data samples and separated train and validation sets.

Data preprocessing pipeline for the training set has the following steps:

- Generate all crop positions for window size 64 with stride 32 along all axis
- Calculate statistics for each segmentation crop (kidney volume, tumor volume)
- Drop empty crops where kidney volume is 0
- Convert voxels of original volume data from NIFTI to greyscale
- Do the min-max scaling of voxels to avoid gradient explosion during the training

2.3 Augmentation

To increase our model generalization capabilities we show more examples to the network by augmenting existed. We consider the following transformations:

- randomly rotate an image in range $[-10°, 10°]$
- randomly scale in range $[0.8, 1.2]$;
- randomly change the brightness in range $[-0.4, 0.4]$;

- we also use both simple random contrast change in range $[-0.4, 0.4]$ and contrast limited adaptive histogram equalization (CLAHE) with probability 0.5.

As one may notice the transformations are very slight, whereas adding the noise, smoothing the image, flip and affine transformations are inappropriate in biomedical imaging.

3 3D Convolutional Neural Network

3.1 Architecture overview

There are many possible architectures suitable for segmentation task but almost all of them are good for 2d segmentation. While we could adopt them to 3d segmentation with slice prediction, lack of context and reduced amount of information can affect the results in a negative way. Because of that we decided to focus on architectures capable to do segmentation of 3D volumes.

Our main choice is 3DUnet [4] which is an adopted version of U-net [12] for 3D volumes. This network is based on the U-shaped architecture, which consists of encoder part to analyze the whole image and compress it to dense representations followed by expanding decoder part to produce a full-resolution segmentation

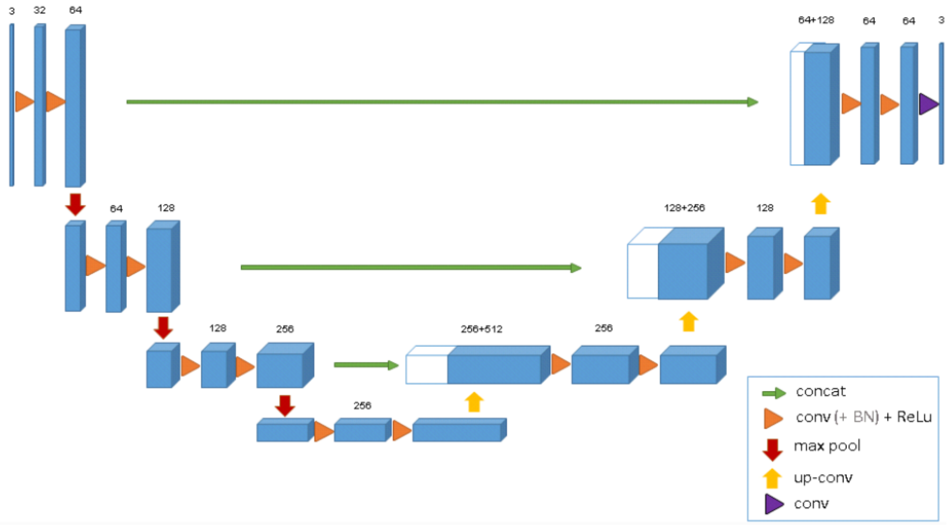


Figure 6: 3D U-Net architecture, taken from [4].

The architecture is pretty much standard. At the encoder part, each layer contains two $3 \times 3 \times 3$ convolutions each followed by a ReLu activation function, and then a $2 \times 2 \times 2$ max pooling with strides of two in each dimension.

In the decoder part, each layer consists of an upconvolution $2 \times 2 \times 2$ with strides of two in each dimension, followed by two 3x3x3 convolutions each followed by a ReLu. Between activation function and convolution layer there is a batch normalization layer. Each batch is normalized during training with its mean and standard deviation and batch normalization parameters are updated using these values.

In the last layer a $1 \times 1 \times 1$ convolution reduces the number of output channels to the number of labels which is 3 in our case

3.2 Training and Testing

The pipeline is quite hard because we are working with very big volumes of data. The main limitation is memory size of the GPU so we have to do some optimization to fit training examples in memory.

3.2.1 Training

To overcome the limitations of the memory size we decided to train our model on small sized crops. This is possible because our neural network is fully convolution and we can have the input of the arbitrary size (as long as we don't rescale it). We have generated positions of crops with size $64 \times 64 \times 64$ with step size equals to 32. It's very important to have overlapping to avoid issues with the prediction on the border. We don't do any data augmentation yet.

As input for our training pipeline, we have a batch of $1 \times 64 \times 64 \times 64$ (here 1 is channel size) crops and on the output, we have a batch of predictions of size $3 \times 64 \times 64 \times 64$. Because we are predicting 3 different classes, we have 3 output channels: for background, kidney and for tumor segmentation.

As a loss function, we used cross entropy function. We train with Adam optimizer [8] for 8 epochs and learning rate 10^{-3}, the minibatch size is 8.

Our training pipeline has the following steps:

- Load data from the hd5f file and prepare crop
- Predict segmentations masks on the crops
- Compute the loss on predictions
- Back propagate and update weights

3.2.2 Testing

To test the prediction we have to implement prediction and score calculation on the predicted. Scoring function (mean of Kidney Sørensen–Dice + Tumor Sørensen–Dice) and is provided by the competition organizers:

$$S = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{2}\left(\frac{2\,n_{t,tp}^{(i)}}{2\,n_{t,tp}^{(i)} + n_{t,fp}^{(i)} + n_{t,fn}^{(i)}} + \frac{2\,n_{k,tp}^{(i)}}{2\,n_{k,tp}^{(i)} + n_{k,fp}^{(i)} + n_{k,fn}^{(i)}}\right)$$

where the first subscript letter indicates tumor or kidney class and the second indicates the prediction type (*tp* for true positive, *fp* – false positive, *fn* – false negative)

Evaluation implementation is a bit tricky. We need to do prediction with sliding window on the whole sample, sum up the predictions and average results in a statistically correct way.

Our evaluation pipeline:

- Prepare the sample crops
- Initialize an empty reference mask of the image size (for storing crop prediction result)
- Iterate over crops, predict and add to the corresponding place in the reference mask
- Calculate entries (number of predictions per pixel)
- Get a mean (by the entries) of the predictions mask

4 Experiments

During the project there were several preprocessing-training-evaluating iterations. First was an early attempt to train 3D U-Net on the full initial training data crops as described in Section 3.2.1.

The amended training data release was done due to mistakes in annotations and there were most of forces put in final training cases. One of the important advantages of the new release was the interpolated data described in Section 2.1 that we decided to work with further. This allowed us to make a correct train-validation split and evaluate the model before testing.

In short we handled the unbalanced data with weighted cross-entropy loss and got the score 0.65, currently we are developing Dice function that is proven to be a good metrics for the loss in such case, and data augmentation to integrate it in the next training iteration.

5 Conclusions

To be covered:

- Overall result and contribution
- Transferring results to other problems
- How the results will influence the decision making process
- Further direction to move

References

[1] Alom, M.Z., Hasan, M., Yakopcic, C., Taha, T.M., Asari, V.K.: Recurrent residual convolutional neural network based on u-net (r2u-net) for medical image segmentation. arXiv preprint arXiv:1802.06955 (2018)

[2] Beier, T., Pape, C., Rahaman, N., Prange, T., Berg, S., Bock, D.D., Cardona, A., Knott, G.W., Plaza, S.M., Scheffer, L.K., et al.: Multicut brings automated neurite segmentation closer to human performance. Nature Methods **14**(2), 101 (2017)

[3] Chen, H., Dou, Q., Yu, L., Qin, J., Heng, P.A.: Voxresnet: Deep voxelwise residual networks for brain segmentation from 3d mr images. NeuroImage **170**, 446–455 (2018)

[4] Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3d u-net: learning dense volumetric segmentation from sparse annotation. In: International conference on medical image computing and computer-assisted intervention, pp. 424–432. Springer (2016)

[5] Ficarra, V., Novara, G., Secco, S., Macchi, V., Porzionato, A., De Caro, R., Artibani, W.: Preoperative aspects and dimensions used for an anatomical (padua) classification of renal tumours in patients who are candidates for nephron-sparing surgery. European urology **56**(5), 786–793 (2009)

[6] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778 (2016)

[7] Heller, N., Sathianathen, N., Kalapara, A., Walczak, E., Moore, K., Kaluzniak, H., Rosenberg, J., Blake, P., Rengel, Z., Oestreich, M., et al.: The kits19 challenge data: 300 kidney tumor cases with clinical context, ct semantic segmentations, and surgical outcomes. arXiv preprint arXiv:1904.00445 (2019)

[8] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)

[9] Kutikov, A., Uzzo, R.G.: The renal nephrometry score: a comprehensive standardized system for quantitating renal tumor size, location and depth. The Journal of urology **182**(3), 844–853 (2009)

[10] Lee, K., Zung, J., Li, P., Jain, V., Seung, H.S.: Superhuman accuracy on the snemi3d connectomics challenge. arXiv preprint arXiv:1706.00120 (2017)

[11] Milletari, F., Navab, N., Ahmadi, S.A.: V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: 2016 Fourth International Conference on 3D Vision (3DV), pp. 565–571. IEEE (2016)

[12] 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, pp. 234–241. Springer (2015)

[13] Taha, A., Lo, P., Li, J., Zhao, T.: Kid-net: convolution networks for kidney vessels segmentation from ct-volumes. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 463–471. Springer (2018)

[14] Zeng, T., Wu, B., Ji, S.: Deepem3d: approaching human-level performance on 3d anisotropic em image segmentation. Bioinformatics **33**(16), 2555–2562 (2017)