SEGMENTATION OF THORACIC ORGANS AT RISK IN CT IMAGES USING LOCALIZATION AND ORGAN-SPECIFIC CNN

Vladimir Kondratenko[1,2,3] *Dmitry Denisenko*[1,2,3] *Artem Pimkin*[1,2,3] *Mikhail Belyaev*[1,2]

[1] Skolkovo Institute of Science and Technology, Moscow, Russia
[2]Kharkevich Institute for Information Transmission Problems, Moscow, Russia
[3]Moscow Institute of Physics and Technology, Moscow, Russia
vladimir.kondratenko@phystech.edu, dmitry.denisenko@skoltech.ru,
artem.pimkin@phystech.edu, m.belyaev@skoltech.ru

ABSTRACT

Segthor19 is the competition timed to the conference IEEE ISBI 2019 that addresses the problem of organs at risk segmentation in Computed Tomography (CT) images. In this paper, we describe our best solution based on convolutional neural networks and challenges that we faced during the competition. Applying this approach we finished on the 24th place in the leaderboard.

***Index Terms*—** Convolutional Networks, Organs segmentation, Computed Tomography (CT) images

1. INTRODUCTION

The problem of organs at risk segmentation rise in the field of radiotherapy. Some approaches allow targeted irradiation of the foci (usually tumour) and their "burn out". In this case, the doctor solves the following problem: he needs to plan the treatment in such a way that:

1. A tumor has a dose of at least a certain threshold.
2. For most of the healthy tissue is not more than a certain (other) threshold.
3. For some critical structures (brain stem, optic nerves, heart, etc.), the dose was no more than one more threshold (as a rule, much less than the threshold from point 2).

To carry out these calculations, they are handing or semi-automatically circling both pockets and critical structures. We can solve the problem with foci more or less well, but with critical structures, the situation is more complicated.

The modern automatic segmentation approaches often use deep neural network structures like U-net[1] and classic computer vision. Hence, these techniques are state-of-the-art; we decided to use them to create our solution. We'll discuss in details the models' structure and parameters selection in the corresponding section of this paper.

2. PROBLEM

The goal of the SegTHOR challenge was to automatically segment 4 OAR: heart, aorta, trachea, esophagus. As a participant, we've been provided with a training set of 40 CT scans with manual segmentation[2]. The organisers then measure the quality on the test set of other 20 CT scans which was published after 1.5 months from the start of the challenge

2.1. Data

We got the train and test data in the standard Nifti-1 file format[1][2][3]. Nifti-1 include not only scan itself but an impressing meta information.

2.1.1. Features and limitations

The typical limitations for biomedical imaging are the small number of training samples and extremely high resolution of the scans (for example matrix of size $512 \cdot 512 \cdot 250$). And also there is a list of natural problems connected with CT scans:

1. People of different height.
2. Different apparatus settings and hence different voxel (3d pixel) sizes.
3. Different time of the equipment start when creating a CT scan.
4. Implants (such as cardiostimulators).
5. Features of the body of each person.
6. Errors in handmade segmentation.
7. The difference in markup by different doctors.

[1]https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3948928/
[2]https://nifti.nimh.nih.gov/nifti-1/
[3]https://brainder.org/2012/09/23/the-nifti-file-format/

2.2. Quality metrics

In competition there are two standard measures of quality:

1. **The overlap Dice metric (DM)**, based on the pixel labeling as the result of a segmentation algorithm, defined as $2 \cdot \frac{|A \wedge B|}{|A|+|B|}$.

2. **The Hausdorff distance (HD)**, defined as $\max(h_a, h_b)$, where h_a is the maximum distance, for all automatic contour points, to the closest manual contour point and h_b is the maximum distance, for all manual contour points, to the closest automatic contour point. The Hausdorff distance is computed in mm thanks to spatial resolution.

Furthermore, in our experiments we also decided to use so-called *surface dice* score[3] which nicely describes the amount of work required for a doctor to fix automatic segmentation. Also, we found it useful for helping us to see if our models perform poorly even without looking at the output. To compute surface dice we used source code from the DeepMind lab repository[4].

3. PROPOSED SOLUTION

In all experiments, we used PyTorch library to create and train models and Dpipe[5] library to manage configurations and run them on our GPU NVIDIA Tesla M40.

3.1. Baseline

3.1.1. Network architecture

We started without any preprocessing using the 2d T-net architecture[4]. We need to introduce the notation of the network configuration which we'll use several times later:

```
[[32, 32],       shortcut(32, 32),       [64, 32, 32]],
[[32, 64, 64],   shortcut(64, 64),    [128, 64, 32]],
[[64, 128, 128], shortcut(128, 128), [256, 128, 64]],
                 [[128, 256, 128]]
```

Numbers in brackets describes the number of filters in each convolution. For example, for the last row it means that we have 128 filters and feed them to convolution with 256 filters, then again apply the convolution and get 128 feature maps. Looking on the architecture scheme is quite useful and convenient for understanding (Fig.1). Unless otherwise stated we use $(3, 3)$ convolutions for the 2d network case and $(3, 3, 3)$ for the 3d network case.

3.1.2. Training

During the training, we picked a random slice from the images and fed them into the network with batch size 8. We also used 5-fold cross-validation to find a sufficient number of epochs to train. After that, we taught this model on the train set of size 22. With this simple approach, we achieved these mean dice scores on the test set gathered from the training data: 0.60 for esophagus, 0.83 for heart, 0.78 for trachea and 0.78 for aorta.

3.1.3. Stacking

We also performed some experiments to check if stacking is helpful for the solution. In first experiments esophagus was predicted much worse than other classes in terms of dice score. So, we decided to add heart, aorta and trachea segmentation in additional channels and feed them to the model. We saw an increase of 0.1 in dice score for esophagus and planned to use this technique for the next experiments with 3d networks.

3.2. Preprocessing

First of all using classic computer vision methods, we removed the medical table and other frames, borders and air. Then we scaled dataset to the same voxel size $0.97 \times 0.97 \times 2.5$ and cut bounding box based on the lungs position (which was received by classic cv methods) of size $(512, 368)$ by x, y from the center and $+20$ slices to top and -125 to bottom by z axis to reduce required computational resources and use bigger batch size. Then we got the outputs from the proposed 2d baseline After that, we performed individual preprocessing for each organ.

3.2.1. Esophagus

We have created the bounding box based on trachea location of size $(128, 128)$ by x, y axes and $+5$ and -115 from the beginning of trachea by z axis.

3.2.2. Trachea

Using the proposed 2d baseline, we took bounding box of size $(128, 128, 60)$ from the 1st slice from the top of the output of the 2d baseline model.

3.2.3. Heart

Using the proposed 2d baseline we took bounding box of size $(512, 368)$ (hence the heart is vast and require a lot of information) by x, y and $+10$ and -50 by z from the 1st slice from the top of the output of the 2d baseline model.

3.2.4. Aorta

Using the proposed 2d baseline, we made two bounding boxes addressed to two different models. Both of them are of size $(128, 184)$ by x, y from the centre. The first one is $+5$ and

[4]https://github.com/deepmind/surface-distance

[5]https://github.com/neuro-ml/deep_pipe

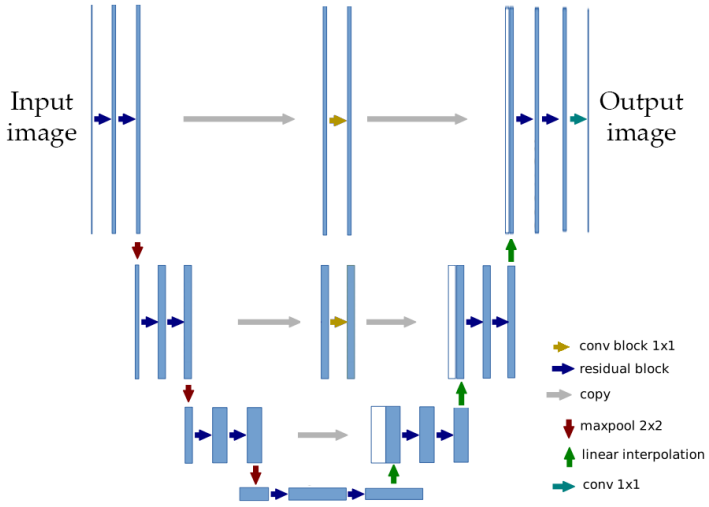

Fig. 1. Example of T-net structure. We use full pre-activation residual blocks from [5]. Conv block is the composition of convolution, Batch Normalization and Relu.

-75 by z and the second one is $+70$ and -10 by z from the first slice from the top of the output of the 2d baseline model.

3.3. Model selection and training

Baseline results show that 2d networks perform not so bad, but for some situations, there is no way they can accomplish nicely. For example, in the top of a human body lungs surround the heart, but in the bottom, there are no similar reference points (see Fig.2). So the spatial information by z axis is required to provide better segmentation. 2d networks can't address this information. Hence, after we performed compression in the preprocessing stage, we are ready to use 3d neural networks as the required amount of computational resources is reduced.

We picked the number of epochs and learning rate using train-test-val split (27-8-5 respectively) and set the batch size to 8.

We found that different organs require different models, mostly because of their geometric sizes. So, we need to entitle these peculiarities[6]:

1. Esophagus: depth 3 T-net with three convolutions (instead of 2) in up-sampling for the better contouring and $(7, 7, 3)$ convolutions in the bottom level. We divided each scan by non-intersecting packages of size 40 by z and fed them into the network.

```
[[16, 16], shortcut(16, 16), [32, 16, 8, 4]],
[[16, 32], shortcut(32, 32), [64, 32, 32, 16]],
      [[32, 64, 128, 128, 64, 32]]
```

2. Trachea: depth 3 T-net with $(7, 7, 3)$ convolutions in the bottom level. We divided each scan by non-intersecting packages of size 20 by z (Fig.3) and fed them into the network.

```
[[16, 16], shortcut(16, 16), [32, 16, 8]],
[[16, 32], shortcut(32, 32), [64, 32, 16]],
      [[32, 64, 128, 128, 64, 32]]
```

3. Heart: depth 3 T-net with seven $(7, 7, 3)$ convolutions in the bottom level. We divided each scan by non-intersecting packages of size 20 by z (Fig.3). The we scaled data before feeding it to the network with the scale factor 0.25.

```
[[16, 16],      shortcut(16, 16),    [32, 16, 16]],
[[16, 32, 32], shortcut(32, 32), [64, 32, 16]],
    [[32, 32, 64, 128, 128, 64, 32, 32]]
```

4. Aorta: depth 3 T-net with three convolutions (instead of 2) in up-sampling for the better contouring and $(7, 7, 3)$ convolutions in the bottom level. We used such a model structure for both top and bottom bounding boxes that we described earlier in the preprocessing part. We divided each scan by non-intersecting packages of size 40 by z. Then we scaled data before feeding it to the network with the scale factor 0.5.

```
[[16, 16], shortcut(16, 16), [32, 16, 8, 4]],
[[16, 32], shortcut(32, 32), [64, 32, 32, 16]],
      [[32, 64, 128, 128, 64, 32]]
```

3.4. Postprocessing

After getting the predictions, we found that for some organs the postprocessing is required: for heart we used the biggest

[6]Note: each network structure below starts with [number of channels, 4, 16] initialization convolution blocks

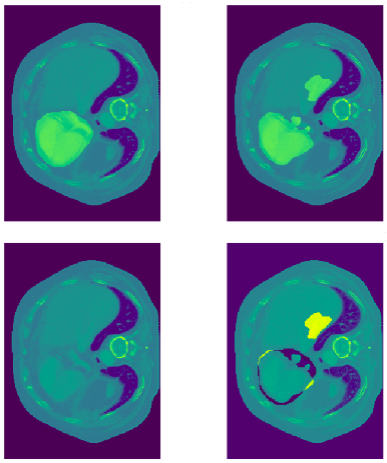

Fig. 2. Example of the lack of spatial information by z for the 2d network (Patient 28). Top left is the ground truth image with the heart segmentation. Top right is the prediction of our 2d network. Bottom left is the image without segmentation, and bottom right is the difference between ground truth image and our prediction. These images demonstrate that the 2d network finds incorrect reference points and start to draw the heart in an inappropriate place.

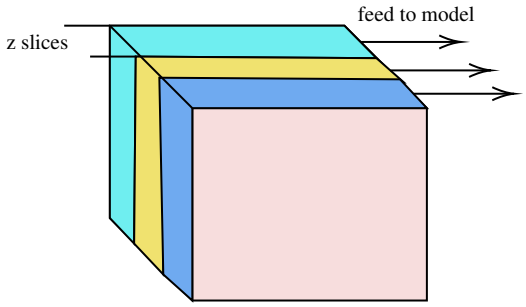

Fig. 3. CT image division on packages

connected component and convex hull, for aorta we took the biggest connected component.

4. FINAL RESULTS AND CONCLUSION

Dice				Hausdorff			
Esophagus	Heart	Trachea	Aorta	Esophagus	Heart	Trachea	Aorta
0.80	0.93	0.89	0.92	0.62	0.30	0.81	0.27

Table 1. Final results

We proposed our solution for organs at risk segmentation in Computed Tomography (CT) images. You can see our results in the Table 1[7]. Moreover, we described our approach to train complicated models on such big data samples with constrained computational resources. We used the non-intersecting packages division of each image. The natural improvement is to use intersecting packages and to take mean values of logits on the reconstruction step or to feed these packages to another network to construct the final output. Also, we planned to use stacking for 3d models. For example, we tried to feed aorta and trachea prediction in additional channels to predict esophagus (hence it worked nicely for the 2d case), but for some reason, this approach performed worse than one described above, so we didn't use it in the final submission.

We also would like to mention the possibility to choose the best segmentation for each patient within previous submissions. The rules of the challenge did not prohibit this, but we decided not to use it since this help to get the highest score but absolutely inapplicable in clinical practice.

The results have been obtained under the support of the Russian Foundation for Basic Research grant 18-29-26030.

5. REFERENCES

[1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[2] Roger Trullo, Caroline Petitjean, Su Ruan, Bernard Dubray, Dong Nie, and Dinggang Shen, "Segmentation of organs at risk in thoracic ct images using a sharpmask architecture and conditional random fields," *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, pp. 1003–1006, 2017.

[3] Stanislav Nikolov, Sam Blackwell, Ruheena Mendes, Jeffrey De Fauw, Clemens Meyer, Cían Hughes, Harry Askham, Bernardino Romera-Paredes, Alan Karthikesalingam, Carlton Chu, et al., "Deep learning to achieve clinically applicable segmentation of head and neck anatomy for radiotherapy," *arXiv preprint arXiv:1809.04430*, 2018.

[4] Artem Pimkin, Gleb Makarchuk, Vladimir Kondratenko, Maxim Pisov, Egor Krivov, and Mikhail Belyaev, "Ensembling neural networks for digital pathology images classification and segmentation," in *International Conference Image Analysis and Recognition*. Springer, 2018, pp. 877–886.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Identity mappings in deep residual networks," 2016.

[7] For trachea we decided to put here the score from our penultimate submission. In last submission the value of the dice score is 0.89 and hausdorff score 0.93