Convolutional neural network stacking for medical image segmentation in CT scans

Marie Kloenne[1]*, Sebastian Niehaus[1]*, Leonie Lampe[1], Alberto Merola[1], Janis Reinelt[1], and Nico Scherf[2,3]

[1] AICURA medical, Bessemerstrasse 22, 12103 Berlin, Germany
`firstname.lastname@aicura-medical.com`
[2] Institute for Medical Informatics and Biometry, Carl Gustav Carus Faculty of Medicine, Technische Universität Dresden, Fetscherstrasse 74, 01307 Dresden, Germany
[3] Max Planck Institute for Human Cognitive and Brain Sciences, Stephanstrasse 1a, 04103 Leipzig, Germany

Abstract. Computed tomography (CT) data poses many challenges to medical image segmentation based on convolutional neural networks (CNNs). The main challenges in handling CT scans with CNN are the scale of data (large range of Hounsfield Units) and the processing of the slices.

In this paper, we consider a framework, which addresses these demands regarding the data preprocessing, the data augmentation, and the CNN architecture itself. For this purpose, we present a data preprocessing and an augmentation method tailored to CT data. We evaluate and compare different input dimensionalities and two different CNN architectures. One of the architectures is a modified U-Net and the other a modified Mixed-Scale Dense Network (MS-D Net). Thus, we compare dilated convolutions for parallel multi-scale processing to the U-Net approach with traditional scaling operations based on the different input dimensionalities. Finally, we combine a set of 3D modified MS-D Nets and a set of 2D modified U-Nets as a stacked CNN-model to combine the different strengths of both model.

Keywords: Medical image segmentation · Computed Tomography (CT) · Kidney tumor segmentation

1 Introduction

Changes in tumor size and shape are important indicators for disease progression and treatment effects. Automated, quantitative assessment of these changes from radiological images would yield a time-saving, efficient, and objective tool for radiologists to monitor the disease course. Thus, a reliable and accurate segmentation method is needed to extract the tumor region from computed tomography (CT) images.

---

* The authors contributed equally to this paper.

In recent years, convolutional neural networks (CNN) became the state of the art method for image classification, image segmentation, object detection, and many other tasks in computer vision. The fields of application are diverse, but the general handling of the data is often very similar, since the feature extraction is done by the CNN. Thus, medical image data, such as CT images, are often processed similarly to other image data. However, this approach neglects prior information about the specific physical processes by which these images are acquired, possibly leading to an inaccurate or suboptimal image analysis. For instance, while most image formats map pixels on relative scales of a few hundred values, voxels in CT volumes are mapped on the Hounsfield scale [2], a quantitative mapping of radiodensity calibrated such that the value for air is -1000 Hounsfield Units (HU) and that for water is 0 HU, with values in the human body reaching up to about 2000 HU (cortical bone). Therefore, in contrast to most natural images where color intensities themselves might not be meaningful, the actual grey values of CT images carry semantic information and special consideration is required to leverage it.

One such consideration is that CT data contain a range of values that are not necessarily relevant for a particular diagnostic question. When radiologists view CT scans for diagnosis, they typically rely on windowing to focus the image information to relevant values. CNN-based image segmentation frameworks rarely include such essential steps from the expert workflow, because it is assumed that the data solely has to be scaled correctly and then a function that focuses on the relevant image regions can be approximated.

In [8], Isensee et al. propose a framework combining data preprocessing and data augmentation, both tailored to medical images needs, with a state-of-the-art CNN. With this framework, they achieved outstanding results in a number of relevant challenges, suggesting that preprocessing and augmentation are indeed critical factors in medical image segmentation and that it is advantageous to leverage prior or domain specific knowledge in these processes.

Building on this experience, in this paper we consider a framework for CNN-based automated kidney tumor segmentation in CT scans, which is trained and validated on the dataset of the 2019 MICCAI KiTS Kidney Tumor Segmentation Challenge [4]. The pipeline is mainly inspired by well known problems in the set up and acquisition of CT volumes and the image-based diagnostic process by radiologists. Differently from the framework proposed in [8], we focus on CT data only. The proposed preprocessing and augmentation refers to a variety in the number and thickness of slices (later referred to as tomography) and the varying positions of patients in the scanner. Tomography is also taken into account during the CNN processing, by evaluating different input dimensionalities for a convolutional neural network with dilated convolutions and for a modified U-Net with traditional scaling operations. The modified U-Net performs better overall, but the predictions of the modified MS-D Net show a lower background error. Therefore, we combine both models to a stacked CNN model.

2   Method

In the following, we describe the data preprocessing and augmentation in section 2.1, the network architectures in section 2.2 and the training procedure in section 2.3. The preprocessing includes especially the volume shape reduction and the intensity windowing, while the proposed augmentation addresses the scarcity of data, with the aim of providing additional data expressions for the training procedure. As for the CNN architectures we consider two solutions: one with dilated convolutions and one with traditional scaling operations. This section also explains how to construct the model stacking. Subsequently, in section 2.3 the training procedure for the two considered architectures is described.

2.1   Preprocessing and Augmentation

In order to ensure an adequate data quality in the training process for each model, we apply a well suited data preprocessing and data augmentation for CT data. The following description of preprocessing is adapted to the dataset of the KiTS Kidney Tumor Segmentation Challenge [4], but can also be easily applied to any other CT dataset.

Image Preprocessing The image normalization is inspired by [8], however changes were implemented to make it more independent from the used data set to enable a more realistic normalization, e.g. for real life applications.

To reduce the complexity, we apply a windowing to each volume by clipping the range to the (0.6, 0.99) percentile range. This represents the range that a radiologist would use for decision-making, but is slightly larger, since no dynamic correction based on the respective CT volume is feasible. For other segmentation problems, this percentile must be adjusted depending on the relevant organs. This windowing is followed by a z-score normalization based on the mean and the standard deviation, which is calculated based on a random sample of the data set.

Since there is a different number of slices per volume, the number of slices needs to be standardized. We decided to reduce each volume to the size of 16 slices, which allows us to modify the volumes without interpolating data. The selection of the slices is random and can be repeated several times per volume, enabling a simultaneous augmentation effect. Background slices are excluded during the training phase, since these are also ignored in the test phase. This step is only relevant if spatial information is to be included. A higher number of slices did not lead to beneficial effects in our experiments, which is explainable by the observation that most convolutional neural networks only use a small semantic context for decision finding [5,11].

To save memory we reduced the shape of each slide by a downsampling using only every fourth voxel. This reduces each slice from a size of 512 x 512 voxels to a size of 128 x 128 voxels.

Image Augmentation In order to train a robust segmentation network, we used different image augmentation techniques. In addition to the extraction of slices already mentioned in the preprocessing description, we also used image rotation, image noising with normal distributed noise map, slice skipping and a range shift. The image rotation simulated patient movement, which cannot be excluded to a certain extent despite fixation, while the other techniques address variances in the CT configuration.

2.2   Architecture

In clinical diagnoses, the localization of tumors is not only done by examining the respective slice, but also the previous and subsequent slices. From a practical point of view, a 3D CNN is therefore preferable in order to avoid losing spatial information. However, previous work has shown that 3D segmentation methods perform worse in anisotropic data [1,7]. Another reason why medical image segmentation with 3D CNNs often proves challenging is the variability in number of slices per volume. The number of slices per volume varies mainly depending on factors like body region under investigation, diagnostic question, variability in subjects' size and acquisition protocol considerations (typically a trade-off between dosing rate, scanning time, and data quality).

To take these challenges and the role of 3D spatial information into account, we consider two different CNN architectures with three different input dimensionalities: a slice-wise 2D segmentation, a multi-channel 2D segmentation (the previous and the following slice are added in two different channels) and a patch-wise 3D segmentation.

The first architecture we consider is a modified U-Net architecture, which is called nnU-Net [8]. This architecture is the native U-Net architecture [3] with instance normalization [14] instead of batch normalization [6] and LeakyReLUs of slope 1e-2 [12] instead of ReLUs. The second architecture we take into account is a modified mixed-scale dense convolutional neural network (MS-D Net) [13], which is modified in the same way as the U-Net. We evaluate the two architectures to test if model performance depends on input dimensionality and whether the data preprocessing and augmentation is working for both the modified U-Nets and the CNN models with dilated convolutions.

Finally we combine different models into a single, stacked CNN model to leverage the different strengths of each architecture. The stacked CNN consist of a set of 3D MS-D Nets, which are trained to detect the kidney without a distinction between the healthy kidney tissue and the tumor tissue, and several 2D nnU-Nets, which are trained to segment all three classes (healthy tissue, tumor and background).

2.3   Training

We trained all networks described in section 2.2 independently and from scratch. The training procedure is shown in Algorithm 1. The procedure is implemented

in Python with Tensorflow 1.14 and performed on the IBM Power System Accelerated Compute Server (AC922).

In each epoch the volumes of a randomly selected batch are preprocessed and augmented, whereby the already mentioned augmentation effect is also achieved with the normalization steps (line 5-10). For the 2D modified U-Net and the 2D

Algorithm 1 Training procedure

1: Initialize network f with random weights θ_0
2: Initialize validation data $V_{validate}$
3: Initialize batch size n
4: **repeat**
5:     **repeat**
6:         Select random volume v
7:         Windowing and Normalization of v
8:         Augmentation of v
9:         Downsampling and slide reduction of v
10:        $V_{batch} \leftarrow v$
11:     **until** Number of v in $V_{batch} = n$
12:     $V_{batch,\hat{y}} = f(V_{batch,x}; \theta_i)$
13:     $L_i = L_{Tanimoto}(V_{batch,\hat{y}}, V_{batch,y})^\alpha + L_{CE}(V_{batch,\hat{y}}, V_{batch,y})^\beta$
14:     $\theta_{i+1} = \text{ADAM}(L_i, \theta_i)$
15:     $L_{validation} = \text{Validate}(f(V_{validate,x}; \theta_{i+1}, V_{validate,y})$
16: **until** Convergence of $L_{validation}$

---

modified MS-D Net a batch size of 28 is used, while for the 3D modified U-Net and the 3D modified MS-D Net the model fits are done with single volumes. In case of 3D segmentation, we augmented 60 percent of training batches and for the training of the 2D models 80 percent of data are augmented, while the range shift is only applied to 20 percent of data.

For the weight update of θ_i for function f, which is a modified U-Net or a modified MS-D Net, we used the ADAM optimization [10] with the proposed parameter configuration and optimized the loss function L (line 13 in Algorithm 1), which is a combination of the Tanimoto loss $L_{Tanimoto}$ and the categorical crossentropy L_{CE}. The weighting of the loss functions is configured for both architectures with $\alpha = 0.6$ and $\beta = 0.4$. Both values are determined experimentally. The Tanimoto loss is implemented as shown in equation 1, where the smooth factor *smooth* is chosen as $1e - 5$. Similar to the well-known Dice score the Tanimoto coefficient treats each class independently, therefore the Tanimoto loss is particularly suitable for problems with a high class imbalance. However, this leads to a class-wise maximum error, if the class does not occur in the sample, which is attenuated by the smooth factor *smooth*. A more detailed discussion is given in [9].

$$L_{Tanimoto}(\hat{Y}, Y) = 1 - \frac{\hat{Y}Y + smooth}{|\hat{Y}|^2 + |Y|^2 - \hat{Y}Y + smooth} \tag{1}$$

3   Evaluation

Both architectures are trained with each data dimensionality as described in section 2.3 and evaluated in a 5 fold cross validation. The prediction are evaluated volume-wise using the Dice score as shown in equation 2. In table 3 the averaged results over all volumes and all folds are shown.

$$s_{Dice}(\hat{Y}, Y) = \frac{2\hat{Y}Y}{|\hat{Y}|^2 + |Y|^2} \tag{2}$$

The results show that the use of 3D spatial information does not necessarily lead to a better segmentation performance and confirm the related work regarding the worst segmentation performance of 3D segmentation methods on anisotropic data. However, the result for the 3D MS-D Net shows the best class-wise score for the background class. This means that for this architecture the whole object (kidney and tumor) is detected well, but the class distinction works comparatively poorly.
Based on these experiments, we constructed the stacked CNN. As described in section 2.2 the stacked CNN consists of a set of 3D MS-D Nets and a set of 2D nnU-Nets. Each set contains the top-5 models. The models are selected based on the validation scored.
This approach yields (i) more stable predictions due to the combinations of different model outputs per architecture and (ii) superior results for the prediction of all three classes due to the combination of two architectures with different individual strengths.

Table 1. Total Dice score (mean $\pm$ std.) in each segmentation class for the different architectures and input dimensionalities (2D, 2D Multi-Channel(M.-C.) and 3D).

		Background	Kidney	Tumor	Total
nnU-Net	2D	0.993 ± 0.004	0.995 ± 0.004	0.994 ± 0.008	0.990 ± 0.008
	2D (M-C)	0.991 ± 0.007	0.992 ± 0.006	0.991 ± 0.007	0.989 ± 0.005
	3D	0.993 ± 0.004	0.991 ± 0.006	0.989 ± 0.007	0.985 ± 0.007
MS-D Net	2D	0.993 ± 0.007	0.989 ± 0.000	0.985 ± 0.000	0.985 ± 0.009
	2D (M-C)	0.993 ± 0.008	0.991 ± 0.007	0.990 ± 0.006	0.989 ± 0.009
	3D	0.997 ± 0.004	0.991 ± 0.008	0.989 ± 0.007	0.989 ± 0.009
Stacked CNN		0.998 ± 0.004	0.999 ± 0.004	0.997 ± 0.005	0.998 ± 0.003

4   Conclusion

In this work, we proposed a machine learning framework for medical image segmentation addressing the specific demands of CT images with respect to pre-processing and data augmentation. We systematically evaluated this framework

for two different state-of-the-art CNN architectures and for different input dimensionalities.

In these experiments, we showed that 3D spatial information does not necessarily lead to positive effects with respect to segmentation performance for fine image structures, which is in line with previous findings. On the other hand, the 3D MS-D Net showed better segmentation results for the background class.

These results suggested that an ensemble-based approach is a good way to combine the complementary strengths of the different combinations of network model and input dimension. We showed that a stacked CNN model indeed outperformed all other approaches in this work as it could leverage the benefits of both architectures by learning a combination of a top-n selection from each model. Our work addresses a central methodological challenge in automated segmentation of CT images. Improved and reliable tumor segmentation is highly relevant to medical diagnosis. Existing clinical nephrometry scores have a poor predictive power [4] and massively reduce the underlying, information contained in CT images. The improved parameterization of kidney tumors through a more efficient, objective, and reliable segmentation, should allow for better clinical evaluation, better prediction of clinical outcomes, and ultimately to a better treatment of the underlying pathology.

References

1. Baumgartner, C.; Koch, L.; Pollefeys, M.; Konukoglu, E.: An Exploration of 2D and 3D Deep Learning Techniques for Cardiac MR Image Segmentation. 2017. arXiv:1709.04496 [cs.CV].
2. Broder, J.: Chapter 9 - Imaging of Nontraumatic Abdominal Conditions, in Diagnostic Imaging for the Emergency Physician. p. 445-577. Elsevier. 2011.
3. Cicek, O; Abdulkadir, A.; Lienkamp, S.; Brox, T.; Ronneberger, O.: 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. Medical Image Computing and Computer-Assisted Intervention (MICCAI), Springer, LNCS, Vol.9901: 424–432. 2015.
4. Heller, N.; Sathianathen, N.; Kalapara, A.; Walczak, E.; Moore, K.; Heather Kaluzniak, H.; Rosenberg, J.; Blake, P.; Rengel, Z.; Oestreich, M.; Dean, J.; Tradewell, M.; Shah, A.; Tejpaul, R.; Edgerton, Z.; Peterson, M.; Raza, S.; Regmi, S.; Papanikolopoulos, N. ; Weight, C.: The KiTS19 Challenge Data: 300 Kidney Tumor Cases with Clinical Context, CT Semantic Segmentations, and Surgical Outcomes. 2019. arXiv:1904.00445 [q-bio.QM].
5. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Wu, E.: Squeeze-and-Excitation Networks. 2017. arXiv:1709.01507 [cs.CV].
6. Ioffe, S.;Szegedy, C.: Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. 2015. arXiv:1502.03167 [cs.LG]
7. Isensee, F.; Jaeger, P.; Full, P.; Wolf, I.; Engelhardt, S.; Maier-Hein, K.: Automatic Cardiac Disease Assessment on cine-MRI via Time-Series Segmentation and Domain Specific Features. 2017.arXiv:1707.00587 [cs.CV].
8. Isensee, F.; Petersen, J.; Klein, A.; Zimmerer, D.; Jaeger, P.; Kohl, S.; Wasserthal, J.; Koehler, G.; Norajitra T.; Wirkert, S.; Maier-Hein, K.: nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation. 2018. arXiv:1809.10486 [cs.CV].

9. Kayalibay B., Jensen G., van der Smagt P.: CNN-based Segmentation of Medical Imaging Data. 2017. arXiv:1701.03056 [cs.CV].
10. Kingma, D.; Ba, J.: Adam: A Method for Stochastic Optimization. 2014. arXiv:1412.6980 [cs.LG]
11. LaLonde, R.; Bagci, U.: Capsules for Object Segmentation. 2018. arXiv:1804.04241 [stat.ML].
12. Maas, A.; Hannun, A.; Ng , A.: Rectifier nonlinearities improve neural network acoustic models. International Conference on Machine Learning (ICML) (2013).
13. Pelt D., Sethian J.: A mixed-scale dense convolutional neural network for image analysis. PNAS 115 (2) 254-259. https://doi.org/10.1073/pnas.1715832114.
14. Ulyanov, D.; Vedaldi, A.; Lempitsky, V.: Instance Normalization: The Missing Ingredient for Fast Stylization. 2016. arXiv:1607.08022 [cs.CV]