

Benchmarking of Deep Architectures for Segmentation of Medical Images

Daniel Gut¹, Zbislaw Tabor, Mateusz Szymkowski, Miłosz Rozynek¹,
Iwona Kucybała¹, and Wadim Wojciechowski

Abstract—In recent years, there were many suggestions regarding modifications of the well-known U-Net architecture in order to improve its performance. The central motivation of this work is to provide a fair comparison of U-Net and its five extensions using identical conditions to disentangle the influence of model architecture, model training, and parameter settings on the performance of a trained model. For this purpose each of these six segmentation architectures is trained on the same nine data sets. The data sets are selected to cover various imaging modalities (X-rays, computed tomography, magnetic resonance imaging), single- and multi-class segmentation problems, and single- and multi-modal inputs. During the training, it is ensured that the data preprocessing, data set split into training, validation, and testing subsets, optimizer, learning rate change strategy, architecture depth, loss function, supervision and inference are exactly the same for all the architectures compared. Performance is evaluated in terms of Dice coefficient, surface Dice coefficient, average surface distance, Hausdorff distance, training, and prediction time. The main contribution of this experimental study is demonstrating that the architecture variants do not improve the quality of inference related to the basic U-Net architecture while resource demand rises.

Index Terms—Benchmark, deep learning, medical image analysis, segmentation.

I. INTRODUCTION

SEGMENTATION of medical images is an important task frequently preceding other image analysis tasks like detection and assessment of abnormalities e.g. tumors or lesions. Given complicated shapes and anatomical variability of internal organs and abnormalities to be segmented as well as a huge variety of textures corresponding to different tissues

Manuscript received 7 March 2022; revised 23 May 2022; accepted 30 May 2022. Date of publication 6 June 2022; date of current version 27 October 2022. This work was supported by the PLGrid Infrastructure [Academic Computer Centre (ACC) Cyfronet Akademia Górniczo-Hutnicza (AGH)]. The work of Daniel Gut was supported by the National Center for Research and Development (NCBR), Poland, under Grant POIR.01.01.01-00-1666/20. (Corresponding author: Daniel Gut.)

Daniel Gut, Zbislaw Tabor, and Mateusz Szymkowski are with the Department of Biocybernetics and Biomedical Engineering, AGH University of Science and Technology, 30-059 Kraków, Poland (e-mail: dgut@agh.edu.pl; ztabor@agh.edu.pl; mateuszszymkowski97@gmail.com).

Miłosz Rozynek, Iwona Kucybała, and Wadim Wojciechowski are with the Department of Radiology, Jagiellonian University Medical College, 31-501 Kraków, Poland (e-mail: miloszrozynek@gmail.com; iwona.kucybała@gmail.com; wadim.wojciechowski@uj.edu.pl).

Digital Object Identifier 10.1109/TMI.2022.3180435

and lesions, as generated by different imaging devices, the segmentation is an extremely difficult problem.

Before the era of deep learning the segmentation problems were solved using ad hoc approaches such that given any specific task very specific algorithms and processing pipelines based on expert knowledge were designed to meet segmentation requirements with respect to its quality. Clearly, such approaches developed for a specific task were not generalizable to other tasks.

Deep learning was certainly the game changer also in the area of medical image segmentation. U-Net architecture, introduced in [1], is an universal segmentation model which since its introduction has been tested on numerous data sets in multiple segmentation tasks [2]. The U-Net architecture is surprisingly simple, given its very good performance. This simplicity naturally raised the question of whether more complicated architectural variants can improve segmentation accuracy without compromising the universality of the modified architectures.

In recent years, many segmentation architectures have been introduced and mentioning all of them certainly exceeds the capacity of a single article. For example, searching for terms “Segmentation” and “U-Net” in abstracts and titles only in PubMed returns almost 1000 articles in the last four years. These articles describe mostly architectures specialized for very specific problems which unfortunately brings us back to the pre-deep learning era of ad hoc approaches. Architectures which are claimed by their authors to be universal and better than U-Net are not so numerous.

To the best of our knowledge, the most influential examples of such universal deep medical segmentation models characterized by a higher complexity than U-Net are UNet++ [3], UNet 3+ [4], ResUNet [5], CS2-Net [6] or CPFNet [7]. Whether this higher complexity indicates better performance is by no means obvious even if the results demonstrated in articles introducing these architectures appear to support such conclusion.

The main problem with these models is that they were never compared with U-Net within a unified framework, which assures that, besides architecture details, other factors which may influence the final results like data preprocessing, a learning rate scheduler, loss function selection, data augmentation etc. are kept fixed. For this reason, in spite of what has been published, it is by no means obvious that any architectural variants are indeed beneficial for medical

image segmentation. Without such a fair comparison it is not clear which architecture should be selected as a first choice whenever facing a new segmentation problem.

In the manuscript, we are using a unified segmentation framework nnU-Net [2] to benchmark recent deep medical segmentation architectures. Because within the framework we control all aspects of deep models training, unlike all other studies focused on the comparison of different architectures, we can separate the effect of a variant of a deep architecture on overall segmentation quality from the effect of all the other factors which may also influence segmentation quality.

From the present study, it follows, in contrast to the conclusions of other studies, that introducing additional complexity to the deep segmentation model, compared to U-Net, does not improve segmentation quality increasing at the same time resource consumption. The codes of the nnU-Net modules with re-implemented deep medical segmentation architectures are available at GitHub: https://github.com/dan-gut/DL_models_benchmark.

The data sets used in this study are also available in the public domain:

- <https://data.mendeley.com/datasets/zm6bxzhmfz> - task 1
- <https://data.mendeley.com/datasets/6x684vg2bg> - task 2
- <https://www.kaggle.com/krzysztofzrecki/bone-marrow-oedema-data> - task 3
- <http://medicaldecathlon.com/> - task 4, 5, 6
- <https://wiki.cancerimagingarchive.net/display/Public/LIDC-IDRI> - task 7
- <https://www.kaggle.com/c/data-science-bowl-2018/data> - task 8
- <https://competitions.codalab.org/competitions/17094> - task 9

II. MATERIALS AND METHODS

A. Materials

The comparison of the performance of deep segmentation architectures is based on nine data sets. These data sets were selected so that they span different modalities (computed tomography (CT), magnetic resonance imaging (MRI), X-rays), different number of input channels, different anatomical regions (head, internal organs, skeleton), different number of foreground objects to be segmented within an image (from 1 to 6), and different types of tissues to be segmented (either normal tissues or pathology). Both custom and open-access data were used in this study. The custom data has been made freely available. Below these data sets are described in detail. Note that besides the object of interest background is always present in all analyzed images and treated as a separate class which should be correctly recognized in the segmented image. Table I shows the number of images in the training and testing sets for each data set.

Data Set 1: The first analyzed data set is a custom data set comprising of X-rays examinations of lower legs performed as a part of routine medical service provided by one of the authors (W.W.) institution. Full-limb X-ray images of 70 randomly selected patients were analyzed. The images were acquired in anteroposterior projection in a standing

TABLE I
COMPARISON OF USED DATA SET SIZES

Task ID	Number of training images	Number of testing images
Task 1	100	40
Task 2	435	140
Task 3	391	162
Task 4	388	96
Task 5	225	55
Task 6	243	60
Task 7	689	146
Task 8	400	64
Task 9	131	23

position with a computed radiography system Philips Digital Diagnost40 (Philips Medical Systems). From these images, rectangular regions corresponding to the left and right hip joints were manually extracted. These regions of interest contained a part of a pelvic bone as well as about 1/3 of the proximal part of a femur. The pixel size of the images was equal to 0.136 mm. The images were coded with contrast resolution equal to 1 byte. Boundaries of pelvic and iliac bones are very clearly visible in X-rays images so they were outlined manually by one experienced radiologist (W.W.) and served as ground truth for training segmentation algorithms expected to segment pelvic and iliac bones as separate objects. From the data set of 140 hip joint regions of interest 100 single-channel images were randomly assigned to a training subset while the remaining 40 single-channel images were assigned to the testing subset. Further details concerning the data set can be found in [8].

Data Set 2: The second analyzed data set is a custom data set comprising of axial slices of abdominal CT scans performed as a part of a diagnostic procedure aimed at the detection of stomach cancer. The slices were selected from full 3D CT abdominal examinations at levels corresponding to the centers of lumbar vertebral bodies L1 to L4. All examinations were performed with the use of a helical 80-row CT scanner Aquilion PRIME 80 (Toshiba America Medical System, Irvine, CA, USA). The pixel size of the images was equal to 0.74 mm while the slice thickness was equal to 5 mm. The images were coded with contrast resolution equal to 2 bytes but, in accordance with DICOM standard, only 12 bits were used to encode signal values with Hounsfield units. Boundaries of spine, spine muscles, abdominal muscles, subcutaneous adipose tissue (SAT), and visceral adipose tissue (VAT) were outlined manually by three radiologists under consensus (initial outlines prepared by I.K. and M.R., verified by I.K., M.R. and W.W.) and served as ground truth for training segmentation algorithms expected to segment separately these regions. Internal organs were also included as separate regions consisting of all pixels which were within the body cross-section but were not included within the aforementioned classes. From the data set of 560 single-channel CT slices 435 images were randomly assigned to a training subset while the remaining 140 images were assigned to the testing subset. Further details concerning the data set can be found in [9].

Data Set 3: The third analyzed data set is a custom data set comprising of coronal oblique slices of MRI examinations of sacroiliac joints. The imaging plane was parallel to the long

axis of sacral bone. 3D MRI examinations of 30 sacroiliac joints were performed as a part of a diagnostic procedure aimed at the detection of axial spondyloarthritis lesions. All examinations were performed with the use of a 3.0 Tesla MRI scanner (Achieva, Philips Healthcare, Amsterdam, Netherlands) and an 8-channel phased-array XL-torso body matrix coil. In this study, only T1-weighted and STIR (short tau inversion recovery) sequences were included. The pixel size of the images was equal to 0.75 mm while the slice thickness was equal to 3 mm. The images were coded with contrast resolution equal to 2 bytes. Boundaries of all bones (iliac and sacral bone) were outlined manually by two radiologists under consensus (initial outlines prepared by I.K., verified by I.K. and W.W.) and served as ground truth for training segmentation algorithms expected to segment bones as foreground possibly disconnected region. The 3D images were split into 2D slices and only slices containing manual segmentations were used for training the segmentation models. From the data set of 553 single-channel MRI slices 391 images were randomly assigned to a training subset while the remaining 162 images were assigned to the testing subset. Further details concerning the data set can be found in [10].

Data Set 4: The data set is a part of the Medical Segmentation Decathlon described there as Task01_BrainTumor. It includes 3D MRI images from patients diagnosed with lower-grade glioma or glioblastoma. These data were used in Brain Tumor Image Segmentation (BraTS) challenge [11]–[13]. Expert board-certified neuroradiologists approved standard annotations of tumor sub-regions [14]. Besides background, the expected result of segmentation should consist of three classes: edema, non-enhancing tumor and signal enhancing tumor. This data set is an example of a multi-channel segmentation problem as at the entrance to the model were four sequences: FLAIR, T1, T2 and T1 after contrast administration. The training data set comprised of 388 randomly selected image tuples while 96 examples were included in the testing data set. The ground truth segmentations are provided within this data set.

Data Set 5: The data set is also a part of the Medical Segmentation Decathlon described there as Task07_Pancreas. It contains a subset of 3D CT images from patients that undergone pancreas masses resection. It is provided by Memorial Sloan Kettering Cancer Center (New York, NY, USA). From the original 420 CT scans obtained in this work 281 images were used [14] of which 225 randomly selected were used for training and the remaining 56 were used for testing. For this single-channel data set the expected segmentation should consist of two classes which are pancreas and cancer.

Data Set 6: The data set is another part of the Medical Segmentation Decathlon described there as Task08_HepaticVessel. It consists of 3D CT images of various liver tumors. It is provided by Memorial Sloan Kettering Cancer Center (New York, NY, USA). 443 CT scans were obtained with an exposure time between 500 and 1100 ms. Each CT image was acquired after iodinated contrast material administration. In this study subset of 303 images were used [14] of which 243 randomly selected were used for training and the remaining 60 were used for testing. For

this single-channel data set the expected segmentation should consist of two classes which are hepatic vessels and hepatic tumor. The ground truth segmentations are provided within this data set.

Data Set 7: The seventh data set is the Lung Image Database Consortium image collection (LIDC-IDRI) [15]. It consists of 1018 CT images of chests. Some of these images were collected for patients with lung cancer in which cases there were lung nodule masks associated with CT examinations. Basing on the lung nodule masks a bounding box was created for each nodule. Then, a padding of 32 voxels was added at each side of the bounding boxes containing nodules and such enlarged patches were extracted from the CT images. In total 835 patches containing nodules were extracted from CT data of which 689 patches were used for training and the remaining ones were used for testing. This data set could not be used with CS2-Net (as it works only with 2D images) and CPFNet (due to the size of images, which is too small to allow such a deep convolution).

Data Set 8: The next data set was provided by the Data Science Bowl 2018 segmentation challenge. This data set consists of nuclei images from different modalities. Images acquired with different modalities substantially differ in appearance. In majority of cases nuclei regions are bright at dark background but there are also images with nuclei regions dark at bright background. For the segmentation task only the images with bright nuclei at dark background were selected. In the original data set each nuclei is assigned a separate label. As the models examined in this article are not designed for instance segmentation, all nuclei regions were assigned the same label. In total 464 images were collected of which 64 were used as testing set. The ground truth segmentations are provided within this data set.

Data Set 9: The last data set used in this study was provided by MICCAI 2017 LiTS Challenge. It consists of 131 CT scans of which randomly selected 100 cases were used for training. The remaining cases were assigned to a test set. At the test time it was found that the size of 8 of the 30 test cases is too large to complete prediction within nnU-Net framework (independently on the architecture for which prediction was run) so finally 23 cases were used for testing. In the ground truth segmentation, there were two different labels corresponding to liver and lesion volumes. For our segmentation task both liver and lesions regions received the same label. The ground truth segmentations are provided within this data set.

B. Architecture Description

Because the influence of non-architectural aspects in segmentation methods could be very impactful, but at the same time it seems to be underestimated, it is extremely important that any comparison between architectures are made under standardized conditions. Recently the hypothesis about the relative importance of non-architectural features for the final success of training deep segmentation models has been stated in [2]. The authors of [2] developed a unified extensible framework nnU-Net for segmenting medical images. The spectacular success of nnU-Net in

several medical image segmentation challenges [16] made us decide that all architecture tested in this benchmark study should be reimplemented as a module of the nnU-Net framework. These reimplementations were based on official code repositories accompanying the articles which introduced the tested architectures. The details about extending nnU-Net framework can be found at github.com/MIC-DKFZ/nnU-Net/blob/master/documentation/extending_nnU-Net.md

The nnU-Net ensures that all architectures are trained on images preprocessed in exactly the same way (including cropping, resampling, and normalization).

Within the nnU-Net framework 5-fold cross validation is used: each training data set is split into 5 subsets, which are then used to train five models for each architecture (in the first training the first subset is used for online validation and the remaining 4 subsets for model parameters learning, in the second training the second subset is used for online validation and the remaining subsets for parameters learning etc.) Note that this split of data sets is the same for each architecture. Each of the five cross-validation trainings was run until convergence on the training subset. During each of the five trainings the best model on the validation subset was saved and used later for inference.

During training, it is assured that:

- the data augmentation is exactly the same for all architectures,
- the optimizer and strategy for changing learning rate is the same,
- the depth of architectures is adjusted to the data in the same way for all architectures,
- the deep supervision is applied in the same way using the same loss,

The inference is also applied in the same way, including ensemble prediction based on the five trained model, which enables estimating variance of the segmentation results.

The codes for re-implementation of the architectures analyzed in this study are available at GitHub. Below we shortly describe the architectures analyzed in this study. The details can be found in the original articles, while implementation details can be found in the codes.

The base model architecture - U-Net - consists of two paths (Fig.1): left contracting - encoder and right expanding - decoder. The encoder consists of stacked layers of base blocks with either a max-pooling operator or stride convolution used for reducing dimensionality. The nnU-Net implementation of U-Net uses strided convolutions. Decoder utilizes transposed convolution operator for increasing dimensionality of the image. Characteristic to U-Net are skip connections between corresponding encoder and decoder layers which are concatenated with output from the previous decoder layer [1].

A variant of U-Net was also considered such that for each U-Net base block there is a residual block processing the data in parallel to this base block. The output of the residual connection is added to the output of the base block. We decided to include ResUNet [5] in this benchmark study motivated by the success of residual connections - based models (ResNet) in classification tasks.

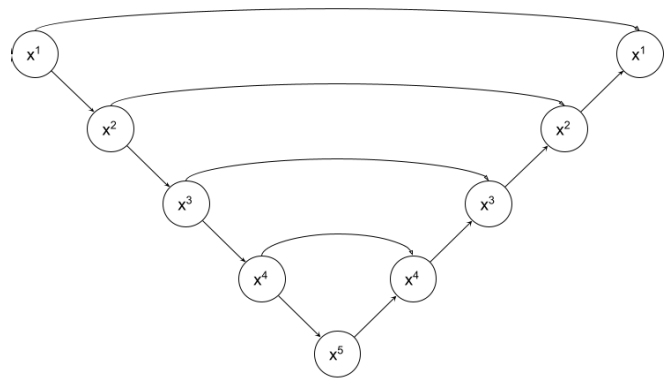


Fig. 1. Visualisation of U-Net architecture.

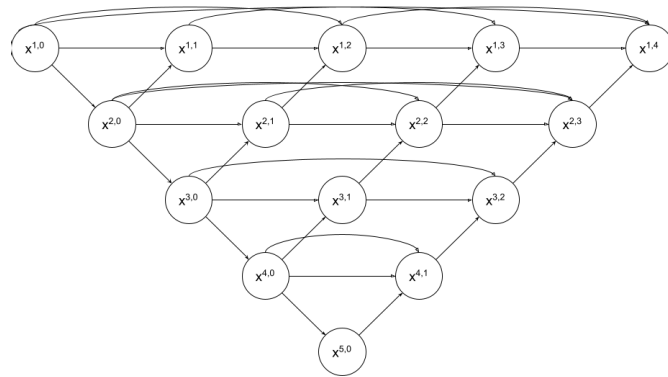


Fig. 2. Visualisation of UNet++ architecture.

The next two architectures: UNet++ and UNet 3+ are U-Net variants in which basic skip connections between the same levels of encoding and decoding parts are replaced by more (UNet++) or less (UNet 3+) complex pattern of skip connections between different levels of encoding and decoding paths of the models. These models were included in this study to test whether skip connection pattern influences segmentation accuracy.

UNet++ [3] architecture is similar to the U-Net, however, it introduces dense skip connections and aggregating blocks, not present in the original U-Net architecture. These aggregating blocks are aimed at aggregating features at varying semantic scales (Fig.2). Like in U-Net, there are direct skip connections between respective layers of encoder and decoder parts. Besides these basic skip connections, there are also skip connections from the encoder to aggregating blocks and other skip connections from aggregating blocks to either other aggregating blocks or decoder layers. The aggregating blocks concatenate features from different scales and then apply convolutions to them before passing them further. The number of aggregating blocks is proportional to the square of the network depth.

UNet 3+ [4] is also based on the U-Net. The skip connections of U-Net are kept in UNet 3+. Like UNet++, UNet 3+ applies some dense pattern of skip connections but, in contrast to UNet++ there are no aggregating blocks (Fig. 3). The aggregation of features at different semantic

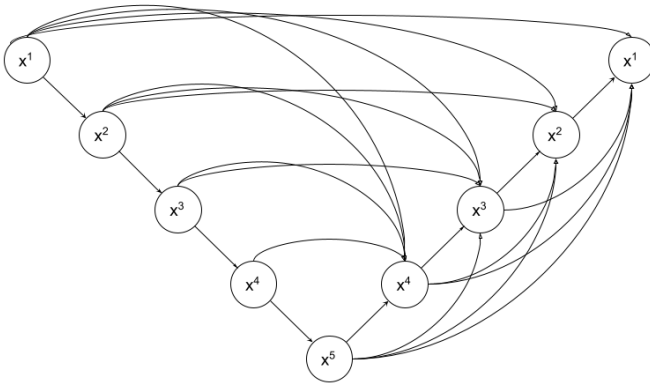


Fig. 3. Visualisation of UNet 3+ architecture.

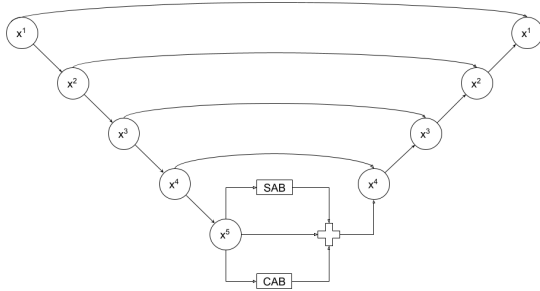


Fig. 4. Figure present architecture of CS2-Net.

scales is accomplished by basic concatenation of features from different layers possibly preceded by appropriate rescaling. Note that the dense skip connections of UNet 3+ are aimed at aggregating features at all scales, that is, there are skip connections even between the top and the bottom layers of encoder and decoder.

CS2-Net in comparison to U-Net modifies the bottleneck of the network, introducing a self-attention mechanism between the encoder and decoder section. It utilizes two attention mechanisms (Fig. 4): spatial attention (SAB) and channel attention (CAB) [6]. In the original paper of [6] basic convolutional blocks of U-Net are replaced by residual convolutional blocks but here we focus only on the influence of the attention mechanism on the overall architecture performance. Note that CS2-Net, due to the design of its attention block, can be applied only to 2D images (or 2D slices of 3D images).

CPFNet [7] is another segmentation model based on the encoder-decoder model which utilizes feature aggregating blocks. In contrast to the UNet 3+, the features are passed upwards from layers of the encoder (Fig. 5). While passing upwards the feature tensors are upsampled and then aggregated in GPG blocks before passing them further to the layers of the decoder. In contrast to the U-Net, the features passed to decoder layers are added to features computed by these layers. According to [7] the purpose of the GPG module is to provide decoder layers with a more global image context. Another feature of CPFNet architecture is a scale-aware pyramid fusion module replacing the bottleneck of U-Net aimed at fusing multi-scale context information at a high level. Note that CPFNet, in contrast to the previous models, has a fixed depth.

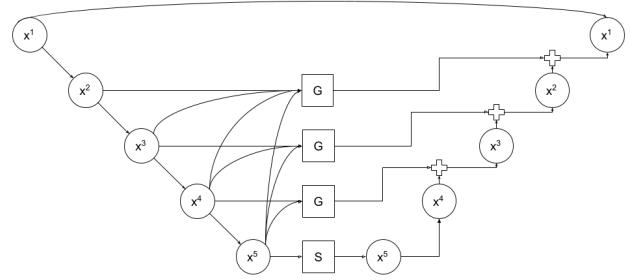


Fig. 5. Architecture of CPF-Net. G denotes GPG blocks and S denotes scale-aware pyramid fusion module.

Note that CPFNet due to the design of its multi-scale context fusion blocks, can be applied only to 2D images (or 2D slices of 3D images).

The U-Net variants selected in this study cover a broad spectrum of possible architecture modifications. Because two architectures are designed only for 2D images, we used 2D variants of all architectures in all tasks, except task 7. It means that for tasks involving 3D data, 3D images were split into sequences of 2D slices and these 2D images were used as model inputs. In the case of task 7, which also involves 3D data, we used 3D variants of architectures (except CS2-Net and CPFNet, designed only for 2D) to check their performance for at least one data set.

C. Metric Description

Metrics that were used for evaluation of performance of trained models:

The number of trainable parameters: The number of trainable parameters was compared as a measure of the complexity and information capacity of each model.

Training time: The mean time of the training epoch for each model was evaluated.

Prediction time: The mean time required to perform a single prediction was measured for each model.

Sørensen–Dice coefficient (DC): The Dice similarity coefficient is a statistical method for determining how similar two sets of data are. It is defined by the equation:

$$QS = \frac{2 * |X \cap Y|}{|X| + |Y|}, \quad (1)$$

where X and Y are two sets, $|X|$ and $|Y|$ are numbers of element in each set, $X \cap Y$ is intersection of this sets [17].

Surface Dice coefficient (SDC): Classical volumetric DC may not give complete insight into quality of segmentation as it doesn't take into consideration the distance from misplaced regions to the segmentation surface. To deal with this issue we used also surface Dice coefficient [18], which allows to assess overlap of two surfaces (at a specified tolerance level, here set to 1). Details of implementation can be found in original article.

Mean surface distance (MSD): Mean surface distance is a measure of distance between points (pixels) p of one surface S to another surface S' (here ground truth segmentation and

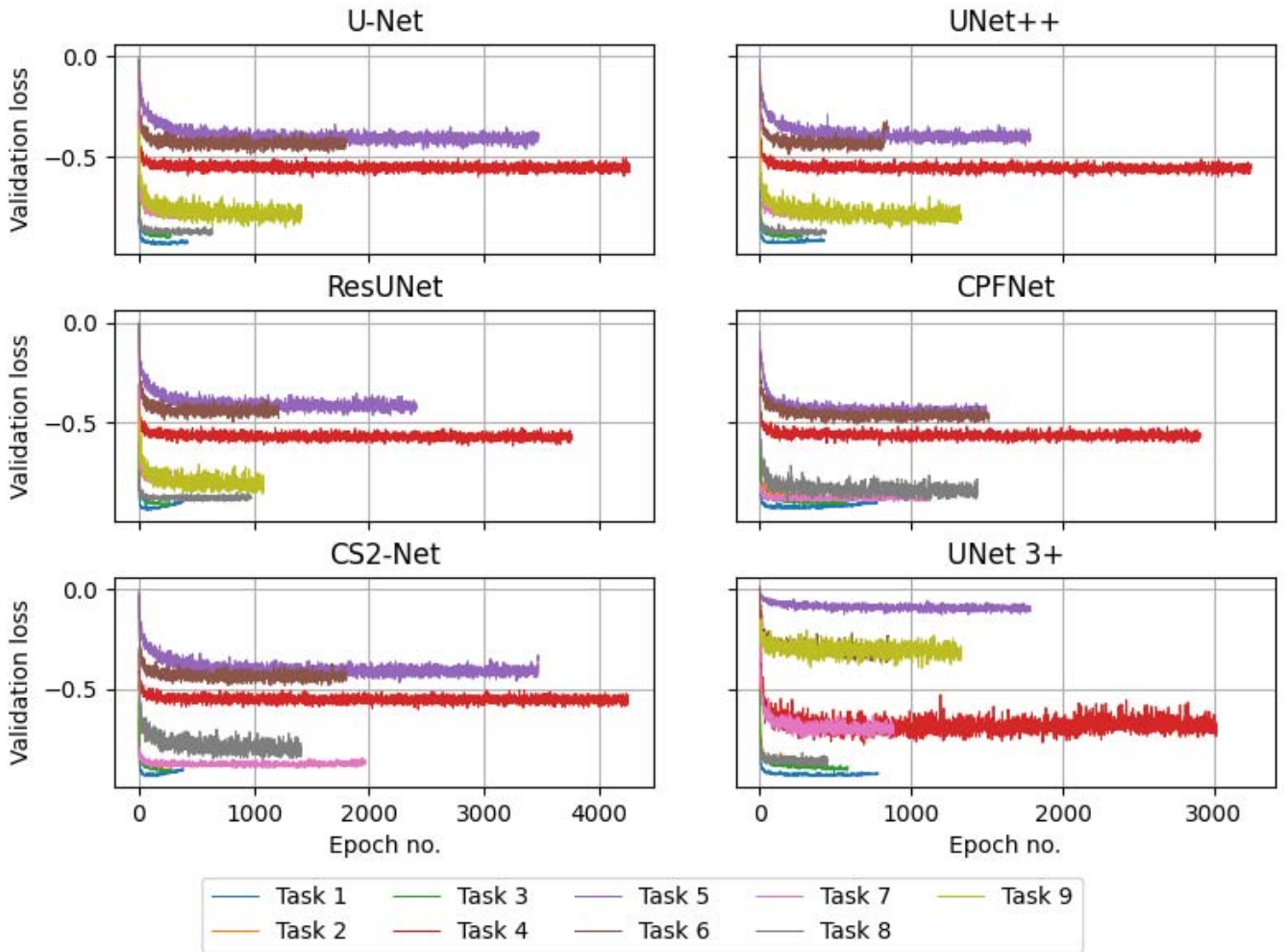


Fig. 6. Value of loss function on validation set during training.

predicted segmentation). Defined as:

$$\text{MSD} = \frac{1}{n_S + n_{S'}} \left(\sum_{p=1}^{n_S} d(p, S') + \sum_{p'=1}^{n_{S'}} d(p', S) \right), \quad (2)$$

where $d(p, S')$ stands for the distance of point p to surface S' defined as minimum of Euclidean norm: $d(p, S') = \min_{p' \in S'} \|p - p'\|_2$. As distances of surface S to S' (ground truth to prediction) and S' to S (prediction to ground truth) are not symmetric, the higher of these two values was used for evaluation of models.

Hausdorff distance (HD): The average Hausdorff distance between two finite point sets X and Y is defined:

$$d_H(X, Y) = \max(\sup_{x \in X} (d(x, Y)) + (\sup_{y \in Y} (d(X, y))), \quad (3)$$

where \sup represents the supremum, \inf the infimum, and $d(x, Y) = \inf_{y \in Y} d(x, y)$ [19].

To compute the aforementioned quality metrics we used python surface-distance package (<https://github.com/deepmind/surface-distance>).

Using the five models trained for each architecture with 5-fold cross validation, five predictions for each architecture

were collected for each testing image. Then, based on these multiple predictions, the standard deviation of each quality metrics has been also estimated for each architecture and each testing image. The standard deviations of quality metrics were then averaged for each architecture over all testing images.

Friedman test was used to test whether there is a difference between the segmentation quality measures computed for the six different architectures. The assumed significance level was equal to 0.05. Whenever the Friedman test indicated a statistically significant difference between models, Nemenyi post hoc tests were run to discover differences between models on a pair-wise basis. Further details concerning application of Friedman and Nemenyi's tests in Machine Learning can be found in [20]. We implemented statistical testing procedures in Python using `scipy.stats` (<https://docs.scipy.org/doc/scipy/reference/stats.html>) and `scikit-posthocs` (https://scikit-posthocs.readthedocs.io/en/latest/posthocs_api/) packages.

III. RESULTS

The training curves for all architectures are shown in Fig. 6. In the figure the loss function values for only validation sets are

TABLE II
NUMBER OF TRAINABLE PARAMETERS

	U-Net	CPFNet	CS2-Net	UNet 3+	UNet++	ResUNet
Task 1, 3	40,803,552	27,922,405	41,207,594	17,777,952	86,313,120	71,617,152
Task 2, 4-9	41,272,032	27,926,245	41,676,074	17,779,744	86,781,600	71,617,152

TABLE III
MEAN TRAINING TIME PER EPOCH IN SECONDS (STANDARD DEVIATION IS GIVEN IN BRACKETS)

	U-Net	CPFNet	CS2-Net	UNet 3+	UNet++	ResUNet
Task 1	101.9 (5.6)	138.7 (1.3)	102.5 (0.8)	405.4 (2.8)	419.2 (4.2)	157.4 (12.5)
Task 2	134.4 (1.4)	177.4 (1.3)	134.5 (8.5)	539.1 (7.5)	544.2 (2.1)	201.8 (1.4)
Task 3	99.4 (0.7)	143.2 (0.8)	99.7 (0.7)	372.3 (7.3)	337.4 (1.4)	157.3 (7.2)
Task 4	52.2 (2.1)	56.1 (0.4)	49.1 (1.0)	82.1 (1.8)	68.6 (0.8)	59.0 (4.2)
Task 5	140.3 (1.4)	178.8 (1.1)	140.1 (3.2)	545.8 (6.3)	538.1 (4.0)	203.7 (10.7)
Task 6	133.2 (0.9)	178.3 (1.3)	133.7 (1.0)	542.6 (7.2)	542.6 (5.0)	201.7 (10.6)
Task 7	485.8 (7.5)	-	-	816.9 (10.6)	1576.0 (15.7)	824.0 (12.1)
Task 8	126.4 (1.8)	317.3 (14.3)	125.4 (1.8)	534.0 (1.1)	543.1 (3.0)	200.44 (5.8)
Task 9	173.2 (6.5)	335.4 (6.1)	174.7 (6.4)	551.1 (11.8)	548.4 (12.7)	226.13 (6.9)

TABLE IV
MEAN PREDICTION TIME IN SECONDS (STANDARD DEVIATION IS GIVEN IN BRACKETS)

	U-Net	CPFNet	CS2-Net	UNet 3+	UNet++	ResUNet
Task 1	0.31 (0.46)	0.66 (0.47)	0.33 (0.47)	2.78 (0.41)	1.22 (0.41)	0.51 (0.50)
Task 2	0.40 (0.49)	0.88 (0.34)	0.41 (0.49)	3.77 (0.42)	1.62 (0.49)	0.60 (0.49)
Task 3	0.30 (0.46)	0.68 (0.46)	0.31 (0.46)	2.63 (0.48)	0.99 (0.07)	0.47 (0.50)
Task 4	7.89 (0.49)	24.63 (0.92)	8.07 (0.49)	52.88 (1.91)	17.47 (0.69)	12.44 (0.60)
Task 5	77.13 (54.28)	191.60 (132.01)	76.97 (54.14)	778.87 (529.77)	358.15 (247.08)	117.82 (87.87)
Task 6	54.39 (53.96)	135.00 (129.26)	54.39 (53.92)	556.84 (513.97)	282.51 (317.21)	86.47 (85.14)
Task 7	5.84 (5.52)	-	-	23.13 (21.26)	19.39 (18.17)	9.99 (9.36)
Task 8	0.41 (0.49)	0.9 (0.29)	0.42 (0.49)	3.79 (0.41)	1.71 (0.45)	0.69 (0.46)
Task 9*	192.93 (140.70)	269.11 (209.45)	196.76 (144.55)	646.56 (573.07)	383.53 (301.55)	232.67 (171.32)

*Due to memory limits predictions for Task 9 were performed on different GPU so absolute value cannot be compared with other tasks.

shown for clarity. All models were trained until convergence on the training set. During the training the models which achieved the best results on validation sets were saved and used for inference on testing sets.

Apart from the ultimate performance of the model, one of its crucial features which were evaluated in this study is the number of trainable parameters and the time required to train it. The more trainable parameters a particular model uses the more information capacity of the model is. The number of trainable parameters in each architecture is compared in Table II. It is slightly dependent on the size of the input images and varies from less than 18 million for UNet 3+ to over 86 million for UNet++. On the other hand, there is no direct relation between the number of trainable parameters and the GPU memory allocated by the framework during the training. In all cases the allocated memory was around 10GB.

A higher amount of additional connections and complexity of neural network strongly influences the number of operations required during training and prediction. It further affects the efficiency expressed as the execution time of a single training epoch (Table III) and the time of a single prediction (Table IV). In both metrics, CS2-Net and U-Net achieved very similar results and were clearly faster than other architectures. In terms of training time on average CPFNet, UNet 3+, UNet++ and

ResUNet were respectively 1.53, 3.37, 3.50 and 1.48 times slower. In the case of the prediction time, the differences are even greater and the average slow-down factor was equal to 2.29, 7.87, 3.65 and 1.56 respectively. It means UNet 3+ required almost 3.5 times more time during training and then almost 8 times more time for each prediction.

Regardless of complexity and time efficiency the ultimate goal of deep learning model is to achieve high performance understood as the ability to correctly segment the image. The Dice coefficient (DC), surface Dice coefficient (SDC), mean surface distance (MSD) and the Hausdorff distance (HD) were calculated for each prediction from the test set made by all architectures, taking into account the division into individual classes in each of the tasks. Tables V to VIII show the medians as well as interquartile range of segmentation quality metrics.

Generally, for task 1, 2, 3, and 8 architectures performed quite good in terms of all quality metrics (for example, the median of DC was higher than 0.9 for all cases, with the differences between architectures not exceeding 0.01). For tasks 7 and 9 the performance was good in terms of some metrics but mild in terms of other metrics, for example DC for Task 9 was high (0.95), but surface DC was small (about 0.7). All models had similarly mild performance for edema and enhancing tumor classes of Task 4 and in Task 7, with median

TABLE V
MEDIAN DICE COEFFICIENT (INTERQUARTILE RANGE IS GIVEN IN BRACKETS)

		U-Net	CPFNet	CS2-Net	UNet 3+	UNet++	ResUNet
Task 1	Thigh bone	0.99 (0.00)	0.99 (0.00)	0.99 (0.00)	0.99 (0.00)	0.99 (0.00)	0.99 (0.00)
	Pelvis	0.97 (0.01)	0.97 (0.01)	0.98 (0.01)	0.98 (0.01)	0.97 (0.01)	0.98 (0.01)
Task 2	SAT	0.99 (0.01)	0.98 (0.02)	0.99 (0.01)	0.99 (0.01)	0.99 (0.01)	0.99 (0.01)
	VAT	0.97 (0.03)	0.96 (0.05)	0.97 (0.03)	0.97 (0.03)	0.97 (0.03)	0.97 (0.03)
	Organs	0.96 (0.01)	0.95 (0.02)	0.96 (0.01)	0.96 (0.02)	0.96 (0.01)	0.96 (0.01)
	Spine	0.97 (0.01)	0.98 (0.01)	0.97 (0.01)	0.97 (0.01)	0.98 (0.01)	0.98 (0.01)
	Spine muscles	0.96 (0.01)	0.97 (0.01)	0.96 (0.01)	0.96 (0.02)	0.96 (0.01)	0.97 (0.01)
	Abdominal muscles	0.95 (0.02)	0.96 (0.01)	0.95 (0.01)	0.95 (0.02)	0.95 (0.01)	0.96 (0.01)
Task 3	Bones	0.93 (0.05)	0.93 (0.05)	0.93 (0.04)	0.93 (0.05)	0.93 (0.04)	0.93 (0.05)
Task 4	Edema	0.83 (0.15)	0.83 (0.14)	0.83 (0.14)	0.77 (0.22)	0.83 (0.15)	0.82 (0.15)
	Non-enhancing tumor	0.61 (0.31)	0.61 (0.34)	0.62 (0.34)	0.52 (0.38)	0.62 (0.32)	0.63 (0.32)
	Enhancing tumor	0.86 (0.13)	0.84 (0.14)	0.85 (0.14)	0.81 (0.17)	0.86 (0.13)	0.85 (0.14)
Task 5	Pancreas	0.75 (0.15)	0.74 (0.13)	0.74 (0.15)	0.67 (0.20)	0.73 (0.14)	0.74 (0.14)
	Tumor	0.35 (0.71)	0.35 (0.58)	0.39 (0.56)	0.16 (0.52)	0.41 (0.53)	0.45 (0.53)
Task 6	Vessels	0.68 (0.10)	0.67 (0.11)	0.68 (0.10)	0.57 (0.25)	0.67 (0.13)	0.68 (0.11)
	Tumor	0.61 (0.45)	0.57 (0.37)	0.63 (0.42)	0.26 (0.45)	0.62 (0.45)	0.63 (0.38)
Task 7	Lesions	0.86 (0.12)	-	-	0.86 (0.15)	0.87 (0.12)	0.86 (0.11)
Task 8	Cells	0.96 (0.06)	0.96 (0.06)	0.96 (0.06)	0.96 (0.07)	0.96 (0.07)	0.96 (0.07)
Task 9	Liver	0.95 (0.02)	0.95 (0.03)	0.95 (0.03)	0.94 (0.03)	0.95 (0.02)	0.95 (0.03)

TABLE VI
MEDIAN SURFACE DICE COEFFICIENT (INTERQUARTILE RANGE IS GIVEN IN BRACKETS)

		U-Net	CPFNet	CS2-Net	UNet 3+	UNet++	ResUNet
Task 1	Thigh bone	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)
	Pelvis	0.99 (0.01)	0.99 (0.01)	0.99 (0.01)	0.99 (0.01)	0.99 (0.01)	0.99 (0.01)
Task 2	SAT	1.00 (0.01)	1.00 (0.01)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)
	VAT	0.99 (0.02)	0.99 (0.02)	0.99 (0.02)	0.99 (0.02)	0.99 (0.02)	0.99 (0.02)
	Organs	0.98 (0.01)	0.99 (0.01)	0.99 (0.01)	0.99 (0.01)	0.99 (0.01)	0.99 (0.01)
	Spine	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)	1.00 (0.00)
	Spine muscles	0.99 (0.01)	1.00 (0.00)	0.99 (0.00)	0.99 (0.01)	1.00 (0.00)	1.00 (0.00)
	Abdominal muscles	0.99 (0.01)	1.00 (0.00)	0.99 (0.00)	0.99 (0.01)	1.00 (0.00)	1.00 (0.00)
Task 3	Bones	0.97 (0.04)	0.97 (0.05)	0.97 (0.04)	0.97 (0.04)	0.97 (0.04)	0.97 (0.04)
Task 4	Edema	0.80 (0.15)	0.80 (0.14)	0.80 (0.14)	0.70 (0.22)	0.81 (0.14)	0.81 (0.15)
	Non-enhancing tumor	0.73 (0.19)	0.70 (0.19)	0.72 (0.20)	0.62 (0.23)	0.71 (0.17)	0.71 (0.17)
	Enhancing tumor	0.91 (0.10)	0.92 (0.12)	0.91 (0.11)	0.86 (0.17)	0.91 (0.12)	0.91 (0.13)
Task 5	Pancreas	0.70 (0.19)	0.68 (0.21)	0.68 (0.17)	0.53 (0.20)	0.67 (0.18)	0.69 (0.20)
	Tumor	0.36 (0.63)	0.33 (0.44)	0.37 (0.51)	0.14 (0.33)	0.34 (0.40)	0.36 (0.40)
Task 6	Vessels	0.82 (0.09)	0.82 (0.09)	0.82 (0.11)	0.74 (0.21)	0.81 (0.13)	0.81 (0.10)
	Tumor	0.47 (0.36)	0.50 (0.36)	0.49 (0.36)	0.23 (0.28)	0.43 (0.33)	0.50 (0.35)
Task 7	Lesions	0.99 (0.06)	-	-	0.99 (0.08)	0.99 (0.06)	0.99 (0.05)
Task 8	Cells	0.99 (0.03)	0.99 (0.03)	0.99 (0.03)	0.99 (0.04)	0.99 (0.03)	0.99 (0.03)
Task 9	Liver	0.70 (0.19)	0.74 (0.13)	0.72 (0.11)	0.68 (0.18)	0.69 (0.13)	0.73 (0.10)

DC above 0.8 (with UNet 3+ performing slightly worse than other models). In the remaining 5 cases (non-enhancing tumor class of Task 4, task 5, and task 6) the performance of the models was poor and the differences between quality metrics were larger.

Statistical significance of the results found for the different architectures was first evaluated with the Friedman test. Given 19 classes in nine tasks and 4 segmentation quality metrics, 76 Friedman tests were run. Using the assumed 0.05 significance level, the null hypothesis about no difference

between the architectures was rejected in 66 out of these 76 cases, which means that in terms of some quality metrics a performance of at least one of the six architectures was different (either better or worse) than the performance of the other architectures. To discover these differences Nemenyi post hoc tests were run (a single post hoc test corresponding to each rejected null hypothesis). Note, that post hoc tests are less powerful than the original tests which means that they may indicate no statistical difference between architectures even if the original test indicated such a difference. Post hoc tests

TABLE VII
 MEDIAN AVERAGE SURFACE DISTANCE IN PIXELS (2D) OR VOXELS (3D) (INTERQUARTILE RANGE IS GIVEN IN BRACKETS)

		U-Net	CPFNet	CS2-Net	UNet 3+	UNet++	ResUNet
Task 1	Thigh bone	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)
	Pelvis	0.1 (0.1)	0.1 (0.1)	0.1 (0.1)	0.1 (0.1)	0.1 (0.1)	0.1 (0.1)
Task 2	SAT	0.0 (0.0)	0.0 (0.1)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)
	VAT	0.1 (0.2)	0.1 (0.3)	0.1 (0.2)	0.1 (0.2)	0.1 (0.2)	0.1 (0.2)
	Organs	0.3 (0.2)	0.1 (0.1)	0.2 (0.2)	0.2 (0.2)	0.2 (0.2)	0.1 (0.1)
	Spine	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)
	Spine muscles	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)
	Abdominal muscles	0.1 (0.1)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)
Task 3	Bones	0.2 (0.7)	0.3 (0.8)	0.3 (0.7)	0.2 (0.7)	0.2 (0.7)	0.3 (0.7)
Task 4	Edema	1.2 (1.1)	1.2 (1.0)	1.2 (1.0)	1.6 (1.9)	1.2 (1.0)	1.3 (1.2)
	Non-enhancing tumor	1.6 (1.8)	1.6 (1.9)	1.5 (2.1)	2.3 (2.4)	1.8 (1.8)	1.8 (1.9)
	Enhancing tumor	0.7 (1.3)	0.7 (0.9)	0.8 (1.3)	1.1 (1.7)	0.9 (1.5)	0.7 (1.0)
Task 5	Pancreas	1.7 (1.2)	1.9 (1.2)	1.9 (1.1)	4.0 (3.6)	1.8 (1.1)	1.7 (1.3)
	Tumor	6.7 (11.6)	7.9 (17.2)	16.2 (21.6)	21.3 (21.9)	12.8 (16.0)	12.3 (21.6)
Task 6	Vessels	2.0 (2.3)	2.1 (1.9)	1.8 (2.2)	3.7 (2.9)	2.0 (2.1)	1.8 (1.8)
	Tumor	15.3 (25.6)	17.3 (31.8)	23.2 (39.5)	48.0 (60.7)	23.1 (45.4)	11.3 (25.8)
Task 7	Lesions	0.2 (0.4)	-	-	0.3 (0.4)	0.2 (0.4)	0.2 (0.3)
Task 8	Cells	0.1 (0.4)	0.1 (0.4)	0.1 (0.4)	0.2 (0.6)	0.1 (0.5)	0.2 (0.5)
Task 9	Liver	5.1 (7.7)	3.5 (6.7)	7.0 (6.6)	12.7 (17.6)	9.0 (7.1)	5.9 (6.6)

TABLE VIII
 MEDIAN HAUSDORFF DISTANCE IN PIXELS (2D) OR VOXELS (3D) (INTERQUARTILE RANGE IS GIVEN IN BRACKETS)

		U-Net	CPFNet	CS2-Net	UNet 3+	UNet++	ResUNet
Task 1	Thigh bone	4.0 (4.2)	4.1 (4.3)	4.0 (4.9)	4.0 (5.2)	3.9 (4.0)	4.0 (4.0)
	Pelvis	11.2 (17.0)	10.5 (14.3)	11.4 (17.1)	9.6 (16.4)	8.3 (14.3)	9.4 (16.8)
Task 2	SAT	8.0 (7.2)	7.2 (8.1)	7.4 (7.9)	8.0 (6.7)	7.2 (8.9)	7.1 (8.2)
	VAT	29.2 (24.5)	27.2 (26.3)	29.9 (22.6)	29.5 (23.6)	29.0 (24.7)	29.8 (21.8)
	Organs	34.9 (18.9)	25.4 (7.5)	32.0 (14.1)	34.1 (19.1)	31.1 (14.6)	28.1 (10.3)
	Spine	3.0 (2.8)	2.2 (1.0)	3.0 (1.8)	3.0 (2.0)	2.5 (1.6)	2.2 (1.0)
	Spine muscles	5.8 (3.3)	4.1 (2.9)	5.7 (3.5)	6.0 (3.1)	5.4 (3.2)	4.1 (2.8)
	Abdominal muscles	7.8 (7.1)	4.1 (3.0)	6.9 (5.5)	7.1 (5.8)	6.1 (5.1)	5.1 (3.2)
Task 3	Bones	14.9 (16.8)	16.9 (17.3)	16.0 (17.6)	15.9 (13.7)	16.4 (16.8)	16.3 (19.5)
Task 4	Edema	25.4 (37.2)	20.3 (22.7)	28.4 (38.0)	21.0 (24.3)	31.8 (36.8)	23.6 (29.0)
	Non-enhancing tumor	22.2 (23.1)	17.5 (16.6)	19.5 (18.9)	18.2 (14.8)	23.9 (24.0)	19.9 (21.0)
	Enhancing tumor	19.8 (16.6)	16.3 (17.4)	20.3 (18.4)	17.8 (14.2)	27.3 (31.2)	19.0 (16.6)
Task 5	Pancreas	16.1 (9.8)	15.2 (13.1)	16.7 (11.8)	34.1 (17.4)	16.9 (9.5)	14.5 (9.2)
	Tumor	22.6 (43.7)	28.4 (63.0)	67.9 (76.7)	84.9 (48.1)	66.9 (64.7)	48.4 (81.9)
Task 6	Vessels	36.9 (70.4)	50.3 (99.9)	37.3 (90.7)	83.8 (107.1)	33.9 (31.2)	35.5 (63.7)
	Tumor	94.6 (81.5)	103.1 (84.2)	153.6 (105.1)	167.3 (101.6)	141.9 (112.8)	89.8 (91.9)
Task 7	Lesions	2.0 (3.1)	-	-	2.0 (3.8)	2.0 (3.1)	1.7 (2.9)
Task 8	Cells	28.4 (69.1)	25.9 (70.0)	27.7 (69.7)	26.0 (70.1)	29.7 (70.8)	31.7 (69.2)
Task 9	Liver	146.5 (115.6)	64.3 (102.4)	146.0 (161.4)	238.3 (206.1)	164.0 (116.8)	128.0 (133.8)

for 66 rejected null hypotheses and 6 architectures returned 990 p-values. To visualize the results of post hoc tests in Fig. 7 we show the differences between quality metrics for U-Net and the competing architectures plotted against the p-values of post hoc tests. Positive values of differences for Dice coefficient or surface Dice coefficient for some architecture mean that the performance of this architecture was worse than that of U-Net in terms of either Dice coefficient or surface Dice coefficient. Analogously, negative values of differences for mean surface distance or Hausdorff distance for some architecture mean that the performance of this architecture was worse than that of

U-Net in terms of either mean surface distance or Hausdorff distance. Clearly, there is no systematic pattern in the figure, that is, neither architecture is consistently better nor worse than U-Net. If some architecture is better than U-Net in some task, it is worse in other tasks. Moreover, even if some architecture is better in terms of some metric than U-Net, the difference in the quality metric may be of no domain significance.

The variability of standard deviation over predictions is consistent with the results reported above, that is, the different architecture variants do not lead to a better performance in terms of e.g. decreased standard deviations over predictions

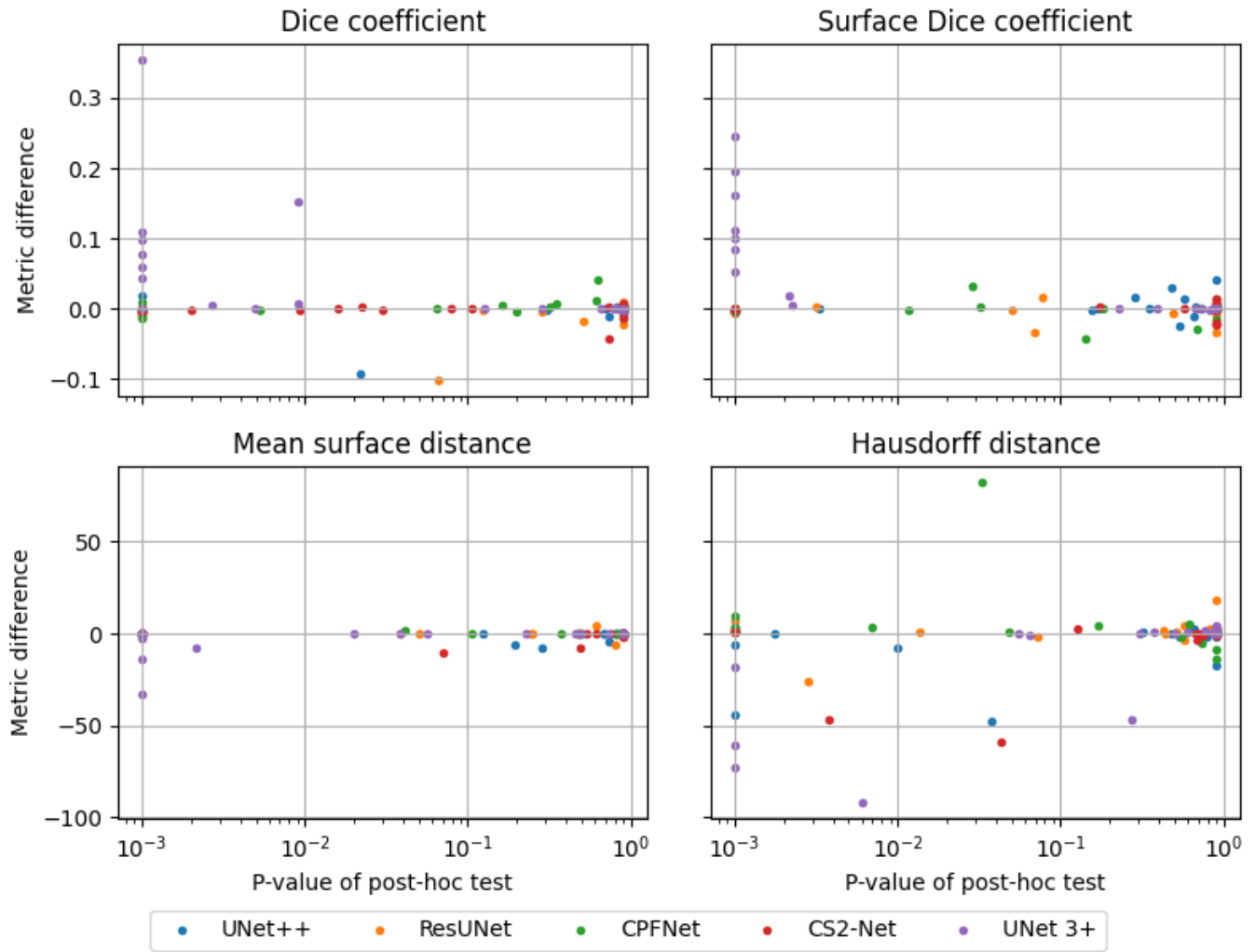


Fig. 7. P-value of post hoc test (comparing to UNet) versus difference of median of metric (comparing to UNet) for different architectures.

made by multiple trained models. For this reason these results are not included here.

IV. DISCUSSION

Since the publication of the U-Net architecture in 2015, there were many propositions on how to improve its performance. The various modifications were compared in this study on nine different segmentation tasks to find out whether there are architectures that are consistently better than others and guaranteeing the best general effectiveness. In this study, the unified nnU-Net framework was used to compare the six selected architectures on a fair basis ensuring that the final results depend only on the architectural detail but not on other factors like data preprocessing, data split into training and testing sets or training strategy. Note, that in accordance with nnU-Net design, the depth of U-Net architecture is always fit to the data before model training according to some rules described in [2]. To ensure a fair comparison, the same depth as computed for U-Net was then used for all the other architectures (besides CPFNet which, by its definition uses fixed depth). In general, it follows from our study that in terms of segmentation quality measures neither of the analyzed models did consistently outperform the classical version of U-Net.

Note that the data sets used in this study can be split into two subsets. For tasks 1, 2, 3, and 8 the segmentation quality metrics are quite high, such that these deep segmentation models can be likely used in clinical practice for diagnostic tasks. While there could be statistically significant differences between segmentation results, as indicated by Friedman's test, these differences are so small that they are of no domain significance. For remaining tasks at least one of the segmentation quality metrics is at most mild and, certainly, not good enough to use these models for diagnostic purposes. For these tasks the differences between models in the terms of any quality metrics, are bigger, which may, potentially, lead to a false conclusions about superiority of one model over another one. As indicated by Nemenyi post hoc tests such conclusions are not supported by the data (for example, due to high variance, differences between median DC of even 10% are not statistically significant, which clearly means that such a difference need not to prove that any of the models is better than others).

Looking at the very similar performance of models the most important factor during the choice of the model is its time and memory efficiency. Complex architectures such as UNet 3+ or UNet++ can be even 3.5 times slower during the training

and almost 8 times slower during prediction than the classical version of U-Net, without any performance improvement. Looking at these performance metrics, classical U-Net and CS2-Net are certainly the best choices.

ACKNOWLEDGMENT

The authors acknowledge the National Cancer Institute and the Foundation for the National Institutes of Health, and their critical role in the creation of the free publicly available LIDC/IDRI Database used in this study.

REFERENCES

- [1] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-24574-4_28#citeas
- [2] F. Isensee *et al.*, "nnU-Net: Self-adapting framework for U-Net-based medical image segmentation," 2018, *arXiv:1809.10486*.
- [3] Z. Zhou *et al.*, "UNet++: Redesigning skip connections to exploit multiscale features in image segmentation," *IEEE Trans. Med. Imag.*, vol. 39, no. 6, pp. 1856–1867, Jun. 2020.
- [4] H. Huang *et al.*, "UNet 3+: A full-scale connected UNet for medical image segmentation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 1055–1059.
- [5] F. I. Diakogiannis, F. Waldner, P. Caccetta, and C. Wu, "ResUNet-A: A deep learning framework for semantic segmentation of remotely sensed data," *ISPRS J. Photogramm. Remote Sens.*, vol. 162, pp. 94–114, Apr. 2020.
- [6] L. Mou *et al.*, "CS2-Net: Deep learning segmentation of curvilinear structures in medical imaging," *Med. Image Anal.*, vol. 67, Jan. 2021, Art. no. 101874.
- [7] S. Feng *et al.*, "CPFNet: Context pyramid fusion network for medical image segmentation," *IEEE Trans. Med. Imag.*, vol. 39, no. 10, pp. 3008–3018, Oct. 2020.
- [8] W. Wojciechowski, A. Molka, and Z. Tabor, "Automated measurement of parameters related to the deformities of lower limbs based on X-rays images," *Comput. Biol. Med.*, vol. 70, pp. 1–11, Mar. 2016.
- [9] I. Kucybała, Z. Tabor, S. Ciuk, R. Chrzan, A. Urbanik, and W. Wojciechowski, "A fast graph-based algorithm for automated segmentation of subcutaneous and visceral adipose tissue in 3D abdominal computed tomography images," *Biocybern. Biomed. Eng.*, vol. 40, no. 2, pp. 729–739, Apr. 2020.
- [10] I. Kucybała, Z. Tabor, J. Polak, A. Urbanik, and W. Wojciechowski, "The semi-automated algorithm for the detection of bone marrow oedema lesions in patients with axial spondyloarthritis," *Rheumatol. Int.*, vol. 40, no. 4, pp. 625–633, Apr. 2020.
- [11] S. Bakas *et al.*, "Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features," *Sci. Data*, vol. 4, no. 1, pp. 1–13, Dec. 2017.
- [12] S. Bakas *et al.*, "Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the BRATS challenge," 2018, *arXiv:1811.02629*.
- [13] B. H. Menze and *et al.*, "The multimodal brain tumor image segmentation benchmark (BRATS)," *IEEE Trans. Med. Imag.*, vol. 34, no. 10, pp. 1993–2024, Oct. 2004.
- [14] A. L. Simpson *et al.*, "A large annotated medical image dataset for the development and evaluation of segmentation algorithms," 2019, *arXiv:1902.09063*.
- [15] S. G. Armato, III, *et al.*, "Data from LIDC-IDRI [data set]," *Cancer Imag. Arch.*, 2015. [Online]. Available: <https://wiki.cancerimagingarchive.net/display/Public/LIDC-IDRI#1966254a2b592e6fba14f949f6e23bb1b7804cc>
- [16] F. Isensee, P. F. Jaeger, S. A. A. Kohl, J. Petersen, and K. H. Maier-Hein, "nnU-Net: A self-configuring method for deep learning-based biomedical image segmentation," *Nature Methods*, vol. 18, no. 2, pp. 203–211, Feb. 2021.
- [17] A. Carass *et al.*, "Evaluating white matter lesion segmentations with refined Sørensen–Dice analysis," *Sci. Rep.*, vol. 10, no. 1, pp. 1–19, Dec. 2020.
- [18] S. Nikolov *et al.*, "Deep learning to achieve clinically applicable segmentation of head and neck anatomy for radiotherapy," 2018, *arXiv:1809.04430*.
- [19] O. U. Aydin *et al.*, "On the usage of average Hausdorff distance for segmentation performance assessment: Hidden error when used for ranking," *Eur. Radiol. Exp.*, vol. 5, no. 1, pp. 1–7, Dec. 2021.
- [20] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, no. 1, pp. 1–30, Dec. 2006. [Online]. Available: <http://jmlr.org/papers/v7/demsar06a.html>