nnU-Net: Breaking the Spell on Successful Medical Image Segmentation

Fabian Isensee[1,2], Jens Petersen[1,3], Simon A. A. Kohl[1*], Paul F. Jäger[1], and Klaus H. Maier-Hein[1]

[1] Division of Medical Image Computing, German Cancer Research Center (DKFZ), Heidelberg, Germany
[2] Faculty of Biosciences, University of Heidelberg, Heidelberg, Germany
[3] Dept. of Neuroradiology, Heidelberg University Hospital, Heidelberg, Germany

Abstract. Fueled by the diversity of datasets, semantic segmentation is a popular subfield in medical image analysis with a vast number of new methods being proposed each year. This ever-growing jungle of methodologies, however, becomes increasingly impenetrable. At the same time, many proposed methods fail to generalize beyond the experiments they were demonstrated on, thus hampering the process of developing a segmentation algorithm on a new dataset. Here we present nnU-Net ('no-new-Net'), a framework that automatically adapts itself to any given new dataset. While this process was completely human-driven so far, we make a first attempt to automate necessary adaptations such as preprocessing, the exact patch size, batch size, and inference settings based on the properties of a given dataset. Remarkably, nnU-Net strips away the architectural bells and whistles that are typically proposed in the literature and relies on just a simple U-Net architecture embedded in a robust training scheme. Out of the box, nnU-Net achieves state of the art performance on six well-established segmentation challenges. Source code is available at https://github.com/MIC-DKFZ/nnunet.

Keywords: Medical Image Segmentation · U-Net · Generalization

1 Introduction

Semantic segmentation remains a popular research topic in the domain of medical image computing with 70% of the international competitions being devoted to it [8]. Important reasons for the enduring interest certainly are the diversity and individual peculiarities of imaging datasets encountered in the medical domain (see e.g. [14] for a detailed overview): datasets vary tremendously when considering cohort size, image dimensionality, image size, voxel intensity ranges and intensity interpretation. Class labels in the images can be highly imbalanced and labels can be ambiguous, while expert annotation quality varies strongly from dataset to dataset. Furthermore, certain datasets are quite inhomogeneous with respect to image geometries or might exhibit slice misalignments and extremely

[*] now with the Karlsruhe Institute of Technology and DeepMind

anisotropic spacings. Taken together, these circumstances make it more difficult to generalize findings from one task to others, and they often prevent immediate success when re-applying methods out of the box to a different problem. The process of adjusting design decisions or proposing new design concepts is complex: most choices are highly dependent on each other, and evidence to substantiate choices is spread over myriads of papers including a lot of "noise". This naturally led to a large number of segmentation methods being proposed in the recent years. Just to provide some prominent examples: variations of encoder-decoder style architectures with skip connections, first introduced by the U-Net [12], include the introduction of residual connections [9], dense connections [6], attention mechanisms [10], additional loss layers [5], feature recalibration [13], and others [11]. The specific modifications differ substantially from each other, but they all share a particular focus on architectural modifications. Given the large amount of segmentation-related publications on the one hand and the diversity of the specific implementations and dataset-related challenges on the other, it becomes increasingly difficult to follow the literature and ascertain which design principles actually generalize beyond the experiments they were demonstrated on. Based on our own experience, many new design concepts did not improve, or sometimes even worsened the performance of a well-designed baseline.

A key problem in medical image computing today is that the process of applying a (segmentation) method to a new problem is completely human-driven. It is based on experience, with the papers mostly focusing on the network architecture, while merely brushing over all the other hyperparameters. Sub-optimal adaptations of a baseline method are regularly compensated for by the proposal of a new architecture. Since the strong dependencies and amount of local minima in hyperparameter space make it really hard to optimally adapt a method to a new problem, nobody in this loop can really be blamed. This situation can be frustrating for the researcher as well as the whole community. Especially in the medical imaging domain where datasets are so diverse, progress will largely depend on our ability to solve these problems.

This paper attempts a first step in this direction: we propose *no-new-Net* (nnU-Net), a segmentation method that includes a formalism for automatic adaptation to new datasets. Based on an automated analysis of the dataset, nnU-Net automatically designs and executes a network training pipeline. Being wrapped around the standard U-Net architecture [12], the hypothesis was that a systematic and careful choice of all hyperparameters will still yield competitive performance. Indeed, without any manual fine-tuning, the method achieves state-of-the-art performance on several well-known medical segmentation benchmarks.

2 Method

A segmentation algorithm can be formalized as a function $f_\theta(x) = \hat{y}$, with x being an image, $\hat{y}$ the corresponding predicted segmentation and θ the set of hyperparameters required for training and applying the method. The dimensionality of θ can be quite large, covering the entire experimental pipeline from preprocessing to inference. Publications usually focus on reporting and substantiating the most relevant choices regarding θ, and ideally provide source code to cover θ entirely. This process, however, lacks insights into how θ must be adjusted if transitioning to a new dataset with different properties. Here, we make the first attempt at formalizing this process. Specifically, we seek for a function $g(X, Y) = \theta$ that generalizes well between datasets. As a first step, this requires identifying those hyperparameters that do not need adaptation, in our case reflecting a strong but simple segmentation architecture and a robust training scheme, and those that are dynamic, i.e. need to be changed in dependence of X and Y. In a second step we define g for the dynamic parameters, which in our case is a set of heuristics rules that adapt the normalization and resampling scheme, configure the patch size and batch size as well as compute the exact network geometries including ensembling and inference techniques. Taken together, this is nnU-Net, a segmentation framework that adapts itself without any user interaction to previously unseen datasets.

2.1 Preprocessing

Image Normalization. nnU-Net requires the information of what modality its input channels are. If a modality is not CT, nnU-Net normalizes intensity values by subtracting the mean and dividing by the standard deviation. If a modality is CT, all foreground voxels in the training set are collected and an automated level-window-like clipping of intensity values is performed based on the 0.5 and 99.5th percentile of these values. To conform with typical weight initialization methods, the data is then normalized with the global foreground mean and standard deviation. The described scheme is independently applied to each case and each modality. **Voxel Spacing.** nnU-Net collects all spacings within the training data and for each axis chooses the median as the target spacing. All training cases are then resampled with third order spline interpolation. Anisotropic spacing (here out-of-plane spacing three times larger than in-plane spacing) can give rise to interpolation artifacts. In this case, out-of-plane interpolation is done using nearest neighbor. For the corresponding segmentation labels, spline interpolation is replaced by resampling each label separately with linear interpolation.

2.2 Training Procedure

Network Architecture. Three U-net models are configured, designed and trained indepentently of each other: a 2D U-Net, a 3D U-Net and a cascade of two 3D U-Net models where the first generates a segmentation at a low resolution which is subsequently refined by the second model. The only notable changes to the original U-Net architecture are the use of padded convolutions to achieve identical output and input shapes, instance normalization and Leaky ReLUs instead of ReLUs. **Network Hyperparameters.** Depending on the shape of the preprocessed training data, the specific instantiation of the U-Nets is adapted.

Specifically, nnU-Net automatically sets the batch size, patch size and number of pooling operations for each axis while keeping the memory consumption within a certain budget (12 GB TitanXp GPU). Hereby, large patch sizes are favored over large batch sizes (with a minimum batch size of 2) to maximize the amount of spatial context that can be captured. Pooling along each axis is done until further pooling would reduce the spatial size of this axis below 4 voxels. All U-Net architectures use 30 convolutional filters in the first layer and double this number with each pooling operation. If the selected patch size covers less than 25% of the voxels in a typical training case, the 3D U-Net cascade is additionally configured and trained on a downsampled version of the training data. The cascade is intended to enable nnU-Net to still acquire sufficient context if the patch size is too small to cover the full resolution. **Network Training.** All U-Net architectures are trained in a five-fold cross-validation. One epoch is defined as processing 250 batches. The sum of the cross-entropy loss and the dice loss are used as loss function. Adam was used as optimizer for stochastic gradient descent with an initial learning rate of 3×10^{-4} and l_2 weight decay of 3×10^{-5}. Whenever the exponential moving average of the training loss does not improve within the last 30 epochs the learning rate is dropped by a factor of 0.2. Training is stopped when the learning rate drops below 10^{-6} or 1000 epochs are exceeded. We apply data augmentation on the fly during training using the *batchgenerators* framework[4]. Specifically, we use elastic deformations, random scaling and random rotations as well as gamma augmentation. If the data is anisotropic, the spatial transformations are applied in-plane as 2D transformations.

2.3 Inference

Cases are predicted using a sliding window approach with half the patch size overlap between predictions. This increases the weight of the predictions close to the center relative to the borders. Test time data augmentation is applied by mirroring along all axes.

nnU-Net ensembles combinations of two U-Net configurations (2D, 3D and cascade) and based on the cross-validation results automatically chooses the best single model or ensemble to be used for test set prediction. For the selected configurations, nnU-Net furthermore uses the five models resulting from the cross-validation for ensembling.

3 Results

nnU-Net was initially developed on the seven training datasets of the phase I from the Medical Segmentation Decathlon challenge [4]. As described in [14], these datasets cover a substantial amount of variability and challenges that are typically encountered in medical segmentation problems. nnU-Net was evaluated both on the Medical Decathlon Segmentation challenge (phase I and phase II)

[4] https://github.com/MIC-DKFZ/batchgenerators/

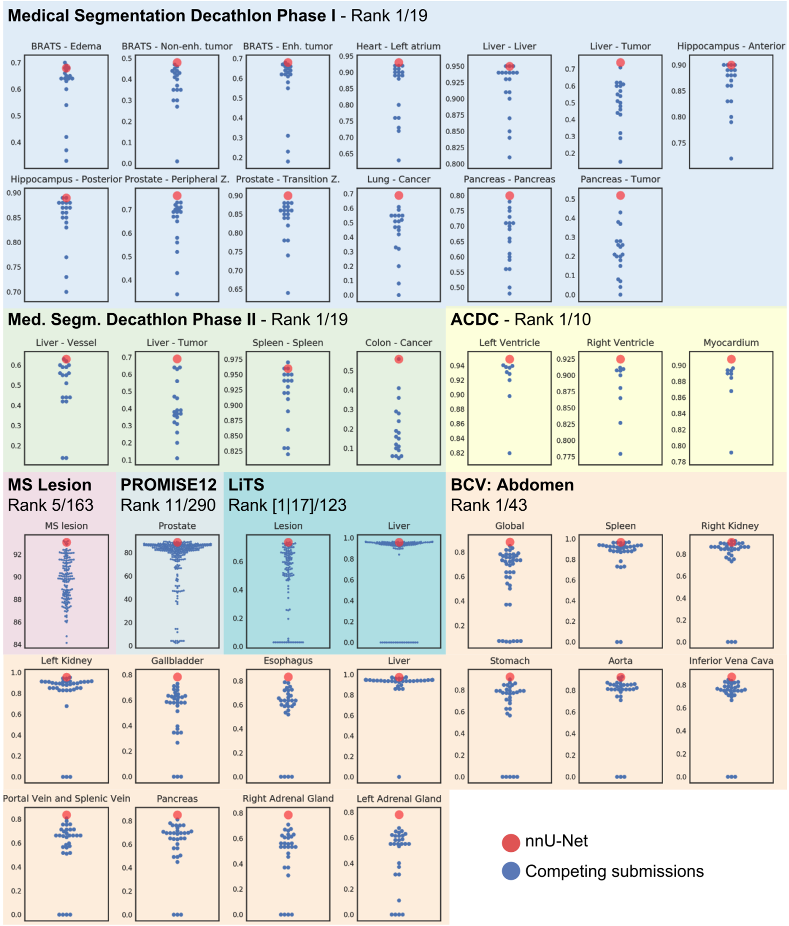
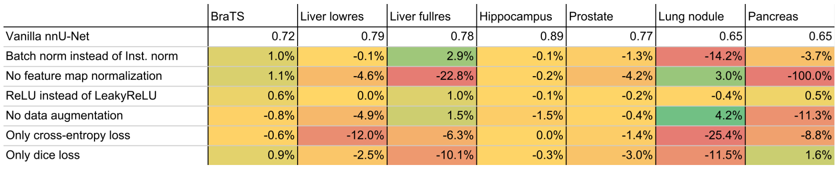


Fig. 1. Summary of nnU-Net performance on the test sets of medical segmentation challenges. All leaderboard submissions are plotted for each dataset and label (accessed on March 25th 2019). Numbers for Decathlon, LiTS, ACDC and BCV are Dice scores, MS lesion and PROMISE12 use different metrics [3,7]. Best viewed in electronic format.

as well as five additional popular medical segmentation challenges. All challenge participations are summarized below and an overview is given in Figure 1.

Medical Segmentation Decathlon (Decathlon). Phase I of this challenge consisted of the seven aforementioned datasets which were used by participants to develop generalizable segmentation algorithms. In phase II, three additional datasets that were previously unknown were made available. Algorithms were applied to these datasets with no further changes or user interaction. The evaluation for both phases was done on the official test sets. nnU-Net was the clear winner of the Decathlon challenge in both phase I and phase II. **Auto-**



	BraTS	Liver lowres	Liver fullres	Hippocampus	Prostate	Lung nodule	Pancreas
Vanilla nnU-Net	0.72	0.79	0.78	0.89	0.77	0.65	0.65
Batch norm instead of Inst. norm	1.0%	-0.1%	2.9%	-0.1%	-1.3%	-14.2%	-3.7%
No feature map normalization	1.1%	-4.6%	-22.8%	-0.2%	-4.2%	3.0%	-100.0%
ReLU instead of LeakyReLU	0.6%	0.0%	1.0%	-0.1%	-0.2%	-0.4%	0.5%
No data augmentation	-0.8%	-4.9%	1.5%	-1.5%	-0.4%	4.2%	-11.3%
Only cross-entropy loss	-0.6%	-12.0%	-6.3%	0.0%	-1.4%	-25.4%	-8.8%
Only dice loss	0.9%	-2.5%	-10.1%	-0.3%	-3.0%	-11.5%	1.6%

Fig. 2. Ablation studies on the design choices of nnU-Net. Experiments were done on representative datasets from the Medical Segmentation Decathlon using one split of the training data and a 3D U-Net. Numerical values for nnU-Net represent the average foreground Dice scores (i.e. mean between liver and tumor dice for the liver dataset), values for the ablation studies represent the percentage-wise change in Dice score.

matic Cardiac Segmentation Challenge (ACDC) [1]: Here, three parts of the heart were to be segmented in cine-MRI images for two different time steps. 100 training cases were provided with two time steps each. We manually split the data for nnU-Nets five fold cross-validation runs to ensure patient stratification. nnU-Net achieved the first place in the open leaderboard (based on the 50 test cases) and set a new state-of-the art on this dataset. **Longitudinal multiple sclerosis lesion segmentation challenge [3]:** The task was to segment MS lesions in MRI images. 5 patients were provided, each with 4-5 time points (21 time points in total) and two raters providing annotations per time point. We treated each rater as a separate training example and again manually split the training cases to ensure patient stratification. On the test set, nnU-Net ranked 5th out of 163 with a score of 93.09 and was just closely behind four submissions from Vanderbilt University of which the highest has a score of 93.21. **PROMISE12 [7]:** The task was the segmentation of the prostate in anisotropic MRI images. 50 annotated training cases and 30 unlabelled test cases were provided. nnU-Net achieved a test set score of 89.08 which places it 11th out of a total of 290 submissions (1st place: 89.59). **LiTS.** The Liver Tumor Segmentation challenge [2] consists of 131 training images (CT) and 70 test cases. For the training cases, segmentations of the liver and liver tumors were provided. nnU-Net achieved dice scores for lesion and liver of 0.725 and 0.958, respectively. Post processing by removing all but the largest connected foreground region improved the dice scores to 0.738 and 0.960, setting a new state of the art in lesion dice while representing the 17th place for the liver label (first place: 0.966) on the open leaderboard (123 teams in total). **Multi-Atlas Labeling Beyond the Cranial Vault Challenge (Abdomen).** Here the task was segmentation of 13 organs in abdominal CT images. The challenge provided 30 annotated training images and 20 test images. nnU-Net set a new state of the art with an average dice score of 88.1%, which is more than 3 points higher than the next best result (43 submission in total on the leaderboard). Looking at the individual organs, nnU-Net scored the highest dice scores in 11/13 organs.

Figure 2 shows ablation studies that were conducted to confirm design choices made in nnU-Net. All experiments were done on one split of the training data and a representative subset of datasets from Phase I of the Decathlon was chosen. These results indicate on the one hand that our decision for using Leaky ReLUs should be revised, but on the other hand confirm our choice of instance normalization, data augmentation and loss function.

4 Discussion

We introduce nnU-Net, a framework that automatically adapts itself to any given medical segmentation dataset without user intervention. To the best of our knowledge, nnU-Net is the first attempt at formalizing the neccessary adaptations that need to be made between datasets. nnU-Net achieves state of the art performance on six publicly available segmentation challenges. This is remarkable given that nnU-Net took away most of the more complicated developments of the recent years and relied on nothing but simple U-Net architectures. Most importantly, it has to be stressed that we did not manually tune hyperparameters between challenges and that all design choices were automatically determined by nnU-Net. It is thus even more surprising that it set new state of the art results in datasets where it specifically competed against manually tuned algorithms.

nnU-Net revolves around both the selection of well-generalizing static design choices such as the U-Net architecture, the dice loss, data augmentation and ensembling, as well as a number of the dynamic design choices that are determined by a set of rules that ultimately reflect our segmentation expertise. Using such rules may however not be the best way to approach this problem. Given a larger number of segmentation datasets, future work could attempt to drectly learn this from the properties of the datasets. While the specific properties of nnU-Net chosen for this publication result in strong segmentation performance across a number of datasets, we do not claim to have found the globally optimal configuration. In fact, looking at the ablation studies presented in Fig. 2, we see that the choice of replacing ReLUs with Leaky ReLUs did not impact performance and that our data augmentation scheme may not be ideal for all datasets. Post-processing should also be investigated further. Our results on LiTS suggest that a properly chosen postprocessing can be beneficial. Such a postprocessing could be automated by analysing the training data or by choosing schemes based on the cross-validation results. An attempt for such automation was part of the initial version of nnU-Net that was used for the Decathlon Challenge, but later discarded for being too conservative and not consistently improving results.

Now that we have established what could be considered the strongest U-Net baseline to date, we can systematically evaluate more advanced network designs with respect to their generalizability as well as their performance gain relative to the plain architecture employed here. nnU-Net can thus not only be used as an out-of-the-box segmentation tool, but also as a strong U-Net baseline and as a platform for future segmentation-related publications. nnU-Net will be made publicly available upon publication at https://github.com/MIC-DKFZ/nnunet.

References

1. Bernard, O., Lalande, A., Zotti, C., Cervenansky, F., Yang, X., Heng, P.A., Cetin, I., Lekadir, K., Camara, O., Ballester, M.A.G., et al.: Deep learning techniques for automatic mri cardiac multi-structures segmentation and diagnosis: Is the problem solved? IEEE TMI **37**(11), 2514–2525 (2018)
2. Bilic, P., Christ, P.F., Vorontsov, E., Chlebus, G., Chen, H., Dou, Q., Fu, C.W., Han, X., Heng, P.A., Hesser, J., et al.: The liver tumor segmentation benchmark (lits). arXiv preprint arXiv:1901.04056 (2019)
3. Carass, A., Roy, S., Jog, A., Cuzzocreo, J.L., Magrath, E., Gherman, A., Button, J., Nguyen, J., Prados, F., Sudre, C.H., et al.: Longitudinal multiple sclerosis lesion segmentation: resource and challenge. NeuroImage **148**, 77–102 (2017)
4. Isensee, F., Petersen, J., Klein, A., Zimmerer, D., Jaeger, P.F., Kohl, S., Wasserthal, J., Koehler, G., Norajitra, T., Wirkert, S., et al.: nnu-net: Self-adapting framework for u-net-based medical image segmentation. arXiv preprint arXiv:1809.10486 (2018)
5. Kayalibay, B., Jensen, G., van der Smagt, P.: Cnn-based segmentation of medical imaging data. arXiv preprint arXiv:1701.03056 (2017)
6. Li, X., Chen, H., Qi, X., Dou, Q., Fu, C.W., Heng, P.A.: H-denseunet: Hybrid densely connected unet for liver and tumor segmentation from ct volumes. IEEE TMI **37**(12), 2663–2674 (2018)
7. Litjens, G., Toth, R., van de Ven, W., Hoeks, C., Kerkstra, S., van Ginneken, B., Vincent, G., Guillard, G., Birbeck, N., Zhang, J., et al.: Evaluation of prostate segmentation algorithms for mri: the promise12 challenge. Med Image Analysis **18**(2), 359–373 (2014)
8. Maier-Hein, L., Eisenmann, M., Reinke, A., Onogur, S., Stankovic, M., Scholz, P., Arbel, T., Bogunovic, H., Bradley, A.P., Carass, A., et al.: Why rankings of biomedical image analysis competitions should be interpreted with care. Nature communications **9**(1), 5217 (2018)
9. Milletari, F., Navab, N., Ahmadi, S.A.: V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: International Conference on 3D Vision (3DV). pp. 565–571. IEEE (2016)
10. Oktay, O., Schlemper, J., Folgoc, L.L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N.Y., Kainz, B., et al.: Attention u-net: learning where to look for the pancreas. arXiv preprint arXiv:1804.03999 (2018)
11. Qin, Y., Kamnitsas, K., Ancha, S., Nanavati, J., Cottrell, G., Criminisi, A., Nori, A.: Autofocus layer for semantic segmentation. In: MICCAI. pp. 603–611. Springer (2018)
12. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: MICCAI. pp. 234–241. Springer (2015)
13. Roy, A.G., Navab, N., Wachinger, C.: Concurrent spatial and channel squeeze & excitationin fully convolutional networks. In: MICCAI. pp. 421–429. Springer (2018)
14. Simpson, A.L., Antonelli, M., Bakas, S., Bilello, M., Farahani, K., van Ginneken, B., Kopp-Schneider, A., Landman, B.A., Litjens, G., Menze, B., et al.: A large annotated medical image dataset for the development and evaluation of segmentation algorithms. arXiv preprint arXiv:1902.09063 (2019)