Learn2Synth: Learning Optimal Data Synthesis Using Hypergradients for Brain Image Segmentation — Supplementary Material —

Xiaoling Hu[1,†], Xiangrui Zeng[1], Oula Puonti[1,2],
Juan Eugenio Iglesias[1,3,4], Bruce Fischl[1,‡], Yaël Balbastre[1,5,‡]

[1]Massachusetts General Hospital and Harvard Medical School
[2]Danish Research Centre for Magnetic Resonance, Copenhagen University Hospital
[3]Centre for Medical Image Computing, University College London
[4]Computer Science and AI Laboratory, Massachusetts Institute of Technology
[5]Department of Experimental Psychology, University College London

In the supplementary material, we begin with the related work in Section 6, followed by the details of the datasets in Section 7 and the experimental details in Section 8. Next, we provide more results in Section 9, Section 10, Section 11 and Section 12, followed by computational cost in Section 13. Finally, we discuss the generalization to 3D in Section 14 and the limitations in Section 15.

6. Related work

Deep learning based medical image segmentation. In the last decades, deep learning methods (CNNs) have provided state-of-the-art accuracy in (medical) image segmentation [3–5, 14–16]. The UNet architecture [16] and its variants [2, 10] has been one of the most popular methods for medical image segmentation. FCN [14] transforms classification CNNs [9, 12, 17] to fully-convolutional NNs by replacing fully connected layers with fully convolutional layers. By doing this, FCN transfers the success of classification tasks [12, 17, 19] to segmentation tasks. Deeplab methods (v1-v2) [3, 5] add another fully connected Conditional Random Field (CRF) after the last CNN layer to make use of global information instead of using CRF as post-processing. Moreover, dilated/atrous convolutions were introduced in Deeplab v3 [4] to increase the receptive field and make better use of context information, resulting in better performance.

While deep learning-based methods have achieved impressive performance metrics, they suffer from two major issues. First, deep learning methods usually require a large amount of high-quality labeled data, which is not realistic in scenarios where domain knowledge is needed to obtain the training data. The second issue is the gaps between different domains. The models trained on one domain do not generalize well to other domains that are different from the training data, which is a major problem in medical imaging due to differences in imaging device vendors, imaging protocols, etc.

To solve both issues aforementioned, in this paper, we propose learning a trainable network to augment the synthetic images, which are then used to train a segmentation network, avoiding the requirements of a large amount of training data and overfitting the training data. Due to the power of UNet for image segmentation with fine structures, in this work, we use UNet as a baseline and our backbone network.

7. Details of the datasets

ABIDE dataset. The Autism Brain Imaging Data Exchange (ABIDE) [6] is a large, publicly available dataset aimed at advancing the understanding of the intrinsic brain architecture in autism. This dataset contains neuroimaging data from individuals with autism spectrum disorder (ASD) as well as typically developing controls, collected from multiple sites over the world. ABIDE includes structural MRI, resting-state fMRI, and other neuroimaging modalities, alongside extensive demographic and clinical information. The whole dataset contains 1087 younger, high-resolution, isotropic, T1 scans.

OASIS3 dataset. The OASIS3 [13] (Open Access Series of Imaging Studies) dataset is a comprehensive, longitudinal collection of neuroimaging, clinical, and cognitive data designed to advance the understanding of normal aging

† Email: Xiaoling Hu (xihu3@mgh.harvard.edu); ‡ Co-senior authors

and Alzheimer's disease (AD). The dataset includes MRI scans, neuropsychological assessments, and clinical evaluations from a diverse cohort of participants, ranging from cognitively healthy individuals to those diagnosed with mild cognitive impairment (MCI) and Alzheimer's disease. The whole dataset contains 1235 older, high-resolution, isotropic, T1 scans.

Buckner39 dataset. The Buckner39 dataset [7] is a comprehensive collection of high-resolution neuroimaging data designed to facilitate the automated labeling and segmentation of neuroanatomical structures within the human brain. The dataset includes T1-weighted MRI scans from 39 healthy adult participants, providing detailed anatomical representations of the brain's major regions. Buckner39 is particularly valuable for evaluating and refining automated segmentation algorithms, offering a benchmark for the accurate identification and labeling of key brain structures across individuals.

Freesurfer maintenance dataset [8] contains images acquired in 8 subjects with a FLASH sequence at multiple flip angles. In contrast with MPRAGE, the FLASH sequence does not apply an inversion pulse, which results in lower cortical contrast. Furthermore, different flip angles give rise to different contrasts. We use FLASH scans acquired with flip angles of 3° and 5° (proton density-weighted) and 20° and 30° (T1-weighted), with the latter being closer in appearance to MPRAGE scans than the former. All scans were skull-stripped and manually delineated with the same protocol as in [7].

Preprocessing of the datasets. The original subjects are in 3D space. We first map the original images and the corresponding masks to the 2D atlas space with the command `mri_convert`: https://surfer.nmr.mgh.harvard.edu/fswiki/mri_convert. Then we use `surfa` (https://surfer.nmr.mgh.harvard.edu/docs/surfa/) package to map all the labels to 4 unique classes.

8. Experimental details

Architecture. We use the standard U-Net [16] as our segmentation backbone. It consists of four resolution levels, where each level contains two convolutional layers (each with a 3×3 convolution followed by a ReLU activation), followed by a 2×2 max pooling operation in the encoder or a transposed convolution (upconvolution) in the decoder.

All experiments use 2D coronal slices extracted from 3D brain MR images. We use the *cornucopia* package* as our synthetic generator and use a standard UNet [16] as the backbone for our segmentation and nonparametric augmentation networks. Architecture details are provided as supplementary material. The networks are randomly initialized and trained from scratch. We use the soft Dice loss [18] to supervise the training of the segmentation network. We use the Adam optimizer [11] with a learning rate of 1×10^{-3}. We apply random intensity augmentations (smoothing, bias field, and noise) and spatial transformations (affine + elastic) to all baselines.

*https://github.com/balbasty/cornucopia

Dataset details for synthetic experiments in Sec. 4.1. We used 60% of the samples as the training set, 20% as the validation set, and the remaining 20% as the test set. At each iteration, noise-free synthetic images are generated from these label maps by assigning random intensities to each label. New images are generated at every epoch, yielding a virtually infinite number of synthetic pairs. Overall results are therefore minimally affected by the portion of training samples.

Dataset details for real-world experiments in Sec. 4.2.1. We used 60% of the samples as the training set, 20% as the validation set, and the remaining 20% as the test set. Both datasets are multicentric; ABIDE aimed to study participants in the autism spectrum, with an age range slightly biased towards younger participants; OASIS aimed to study patients with dementia, with an age range slightly biased towards older participants. Both datasets mostly contain images acquired with an MPRAGE sequence, which is T1-weighted and optimized for cortical contrast.

Dataset details for real-world experiments in Sec. 4.2.2. We use FLASH scans acquired with flip angles of 3° and 5° (proton density-weighted) and 20° and 30° (T1-weighted), with the latter being closer in appearance to MPRAGE scans than the former. All scans were skull-stripped and manually delineated with the same protocol as in [7]. The synthetic portion of the training set used label maps derived from the ABIDE and OASIS3 datasets.

9. More results for the generalizability of *Learn2Synth*

To further quantify the impact of this validation step, we also provide results obtained with fully converged models as well as the best models selected by using the validation set in Table 7.

By comparing the performances of using the validation set or not, we have the following observations:

1. There are significant dice point differences on FLASH data between 'Best' and 'Last' under the same setting, suggesting we will need a validation set for the supervised method.
2. Our proposed *Learn2Synth* is essentially insensitive to using a validation set or not. So *Learn2Synth* only really needs 5 label examples to achieve satisfactory performance, instead of 10 (5 training samples plus 5 validation samples).

Setting	# of Train, Val., Test	Test Set (MPRAGE)	3°	5°	20°	30°
SynthSeg	/	0.861 ± 0.028	0.776 ± 0.012	0.694 ± 0.027	0.766 ± 0.018	0.781 ± 0.016
Supervised UNet (w/ validation)	29, 5, 5	$\mathbf{0.941 \pm 0.002}$	0.419 ± 0.025	0.396 ± 0.020	0.671 ± 0.026	0.769 ± 0.022
Supervised UNet (w/o validation)	29, 5, 5	0.936 ± 0.004	0.301 ± 0.024	0.314 ± 0.022	0.527 ± 0.028	0.637 ± 0.035
Supervised UNet (w/ validation)	5, 5, 29	0.907 ± 0.007	0.397 ± 0.018	0.413 ± 0.016	0.586 ± 0.033	0.692 ± 0.033
Supervised UNet (w/o validation)	5, 5, 29	0.900 ± 0.011	0.342 ± 0.019	0.379 ± 0.020	0.637 ± 0.032	0.728 ± 0.023
Supervised UNet (w/ validation)	1, 5, 33	0.885 ± 0.010	0.267 ± 0.019	0.247 ± 0.013	0.544 ± 0.026	0.651 ± 0.025
Supervised UNet (w/o validation)	1, 5, 33	0.871 ± 0.013	0.337 ± 0.027	0.372 ± 0.032	0.654 ± 0.018	0.727 ± 0.017
Learn2Synth (w/ validation)	29, 5, 5	0.895 ± 0.011	$\mathbf{0.804 \pm 0.014}$	$\mathbf{0.789 \pm 0.010}$	0.785 ± 0.025	0.797 ± 0.023
Learn2Synth (w/o validation)	29, 5, 5	0.897 ± 0.012	0.801 ± 0.012	0.787 ± 0.011	0.782 ± 0.024	0.794 ± 0.023
Learn2Synth (w/ validation)	5, 5, 29	0.867 ± 0.030	0.798 ± 0.013	$\mathbf{0.789 \pm 0.010}$	$\mathbf{0.795 \pm 0.026}$	$\mathbf{0.799 \pm 0.024}$
Learn2Synth (w/o validation)	5, 5, 29	0.864 ± 0.031	0.800 ± 0.014	0.788 ± 0.012	0.786 ± 0.024	0.793 ± 0.023
Learn2Synth (w/ validation)	1, 5, 33	0.718 ± 0.045	0.606 ± 0.020	0.603 ± 0.017	0.690 ± 0.036	0.707 ± 0.032
Learn2Synth (w/o validation)	1, 5, 33	0.725 ± 0.037	0.588 ± 0.024	0.591 ± 0.018	0.695 ± 0.035	0.715 ± 0.031

Table 7. Performance comparison of different models across various datasets.

10. Unpaired segmentation

An advantage of SynthSeg [1] is that it performs unpaired segmentation, which only requires a set of segmentation maps for MRI synthesis during segmentation training. In contrast, our proposed *Learn2Synth* framework requires a small amount of labeled real data during training to learn the synthesis process.

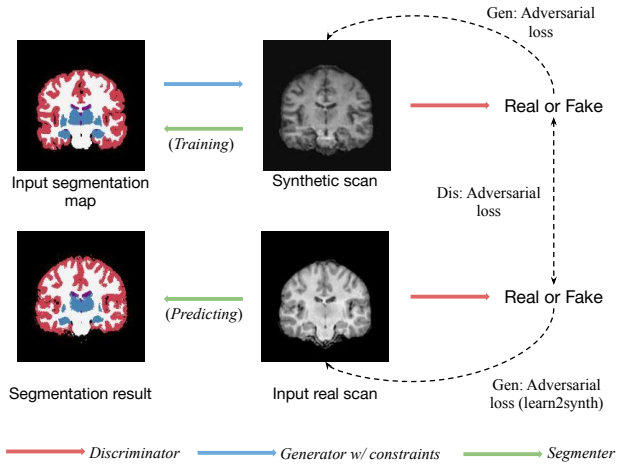

Figure 5. Illustration of the unpaired approach for learning segmentation with *Learn2Synth*. The generator is trained with a unique adversarial loss that simultaneously aims to make synthetic scans more realistic while making real scans appear less authentic to the discriminator through *Learn2Synth* hypergradient back-propagation. This approach does not require paired segmentation-label data, as the input segmentation maps and real scans are independent of each other.

Here, we explore the potential of unpaired segmentation using *Learn2Synth* in combination with a Generative Adversarial Network (GAN). As illustrated in Figure 5, we use a GAN to generate a synthetic MRI scan that is constrained to an input segmentation map. Unlike conventional GANs, the adversarial loss on the generator is applied not only to the synthetic scan but also to the real scan. Specifically, the generator is trained to fool the discriminator by making the fake data appear more realistic, while simultaneously making the real data appear faker, through the *Learn2Synth* hypergradient back-propagation. The input segmentation maps and real scans are independent of each other, meaning that paired segmentation-label data is not required.

We conducted experiments under the same conditions as described in Section 4.2.1. Unpaired segmentation maps from ABIDE and OASIS-3, along with scans from ABIDE, OASIS-3, and Buckner39, were used as training samples. The trained segmenter was then applied to the Buckner39 dataset, resulting in a Dice score of 0.874 ($\pm$ 0.0104), outperforming SynthSeg (0.861), as shown in Table 6.

11. nnUNet as backbone

Table 8 shows results using nnUNet as the seg. backbone for both 'Naive SynthSeg' and *Learn2Synth*, with improved performance but much longer training time ($\approx$62.1h vs $\approx$17.7h on OASIS3). Our focus is on training strategies rather than architectures; networks can be treated as black boxes, and contrast invariance cannot be learned if trained only on T1w images.

Method	ABIDE	OASIS3
Naive SynthSeg	0.881	0.863
Learn2Synth (parametric setting fixed σ)	**0.893**	**0.875**

Table 8. Comparison using nnUNet as seg. backbone.

12. Comparison with the same test size and add Mixed SynthSeg as baseline

We also reported the results in Table 6 (main text) using a consistent test set size across all comparisons for interpretable evaluation. Additionally, we have included both the 'Mixed SynthSeg' and 'Finetuned SynthSeg' baselines for comparison (Table 9). We only include the results here for the flip angle of FLASH 3° for space limitations, and the others will be included in the revised version.

Setting	# of Train, Val., Test	Test Set (MPRAGE)	3°
SynthSeg	/	0.861 ± 0.028	0.776 ± 0.012
Mixed SynthSeg	5, 5, 5	0.859 ± 0.021	0.781 ± 0.013
Finetuned SynthSeg	5, 5, 5	0.863 ± 0.017	0.783 ± 0.015
Supervised UNet	29, 5, 5	$\mathbf{0.941 \pm 0.002}$	0.419 ± 0.025
Supervised UNet	5, 5, 5	0.910 ± 0.012	0.397 ± 0.018
Supervised UNet	1, 5, 5	0.879 ± 0.014	0.267 ± 0.019
Learn2Synth	29, 5, 5	0.895 ± 0.011	$\mathbf{0.804 \pm 0.014}$
Learn2Synth	5, 5, 5	0.871 ± 0.028	0.798 ± 0.013
Learn2Synth	1, 5, 5	0.725 ± 0.019	0.606 ± 0.020

Table 9. Comparison of models across various datasets.

13. Computational cost and framework complexity

For OASIS3, taking the '*Learn2Synth* (parametric setting with fixed σ)' as an example, the model converges after 1,500 epochs with a batch size of 64, requiring $\approx$17.7 hours of training time on an NVIDIA L40S GPU (48GB), using a 64-core Intel(R) Xeon(R) Gold 6438Y+ CPU and 200 GB RAM. For comparison, 'Naive SynthSeg' requires $\approx$10.3 hours under the same setup.

While *Learn2Synth* introduces alternating synthetic and real passes, it eliminates the need for costly cross-validation typically used to tune augmentation hyperparameters. Unlike grid search, which scales exponentially with the number of parameters, *Learn2Synth* uses hypergradient-based updates to optimize parameters in a single training run, making the process far more efficient and scalable.

A key advantage of *Learn2Synth* is that it avoids manual hyperparameter tuning by treating augmentation parameters as learnable variables. Unlike traditional methods like SynthSeg that rely on heuristic tuning, *Learn2Synth* uses hypergradients and a small set of real validation data to automatically optimize augmentations, improving generalizability across domains.

14. Generalization to 3D data

While our experiments use 2D slices, the *Learn2Synth* framework is architecture- and dimensionality-agnostic and readily extends to 3D. Its synthetic-to-real training and hypergradient-based optimization remain applicable in volumetric settings, which pose challenges like anisotropy and high memory demands. We plan to explore full 3D evaluations in future work.

15. Limitations

One limitation compared with naive SynthSeg [1] is that we need labeled real scans in the target modality to optimize the segmentation results. Also, this work focuses specifically on segmentation, as indicated by the paper's scope. While the core ideas of *Learn2Synth* could extend to tasks like registration or lesion detection, we intentionally limit our study to segmentation for a focused evaluation. Exploring other tasks is left for future work.

References

[1] Benjamin Billot, Douglas N Greve, Oula Puonti, Axel Thielscher, Koen Van Leemput, Bruce Fischl, Adrian V Dalca, Juan Eugenio Iglesias, et al. SynthSeg: Segmentation of brain MRI scans of any contrast and resolution without retraining. *MedIA*, 2023. 3, 4

[2] Jieneng Chen, Yongyi Lu, Qihang Yu, Xiangde Luo, Ehsan Adeli, Yan Wang, Le Lu, Alan L Yuille, and Yuyin Zhou. Transunet: Transformers make strong encoders for medical image segmentation. *arXiv preprint arXiv:2102.04306*, 2021. 1

[3] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with deep convolutional nets and fully connected CRFs. *arXiv preprint arXiv:1412.7062*, 2014. 1

[4] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 1

[5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *TPAMI*, 2018. 1

[6] Adriana Di Martino, Chao-Gan Yan, Qingyang Li, Erin Denio, Francisco X Castellanos, Kaat Alaerts, Jeffrey S Anderson, Michal Assaf, Susan Y Bookheimer, Mirella Dapretto, et al. The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism. *Molecular psychiatry*, 2014. 1

[7] Bruce Fischl, David H Salat, Evelina Busa, Marilyn Albert, Megan Dieterich, Christian Haselgrove, Andre Van Der Kouwe, Ron Killiany, David Kennedy, Shuna Klaveness, et al. Whole brain segmentation: automated labeling of neuroanatomical structures in the human brain. *Neuron*, 2002. 2

[8] Bruce Fischl, David H Salat, André JW Van Der Kouwe, Nikos Makris, Florent Ségonne, Brian T Quinn, and Anders M Dale. Sequence-independent segmentation of magnetic resonance images. *Neuroimage*, 2004. 2

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1

[10] Fabian Isensee, Paul F Jaeger, Simon AA Kohl, Jens Petersen, and Klaus H Maier-Hein. nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature methods*, 2021. 1

[11] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 2

[12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, 2012. 1

[13] Pamela J LaMontagne, Tammie LS Benzinger, John C Morris, Sarah Keefe, Russ Hornbeck, Chengjie Xiong, Elizabeth Grant, Jason Hassenstab, Krista Moulder, Andrei G Vlassenko, et al. OASIS-3: longitudinal neuroimaging, clinical, and cognitive dataset for normal aging and alzheimer disease. *medrxiv*, 2019. 1

[14] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 1

[15] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *ICCV*, 2015.

[16] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 1, 2

[17] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[18] Carole H Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017*, 2017. 2

[19] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 1