

# Domain Generalizer: A Few-shot Meta Learning Framework for Domain Generalization in Medical Imaging

Pulkit Khandelwal<sup>1,2</sup> ✉ and Paul Yushkevich<sup>2</sup>

<sup>1</sup> Department of Bioengineering, University of Pennsylvania, Philadelphia, PA, United States

<sup>2</sup> Penn Image Computing and Science Laboratory, Department of Radiology, University of Pennsylvania, Philadelphia, PA, United States  
pulks@seas.upenn.edu

**Abstract.** Deep learning models perform best when tested on target (test) data domains whose distribution is similar to the set of source (train) domains. However, model generalization can be hindered when there is significant difference in the underlying statistics between the target and source domains. In this work, we adapt a domain generalization method based on a model-agnostic meta-learning framework [1] to biomedical imaging. The method learns a domain-agnostic feature representation to improve generalization of models to the unseen test distribution. The method can be used for any imaging task, as it does not depend on the underlying model architecture. We validate the approach through a computed tomography (CT) vertebrae segmentation task across healthy and pathological cases on three datasets. Next, we employ few-shot learning, i.e. training the generalized model using very few examples from the unseen domain, to quickly adapt the model to new unseen data distribution. Our results suggest that the method could help generalize models across different medical centers, image acquisition protocols, anatomies, different regions in a given scan, healthy and diseased populations across varied imaging modalities.

**Keywords:** domain adaptation · domain generalization · meta learning  
· vertebrae segmentation · computed tomography

## 1 Introduction and Background

In biomedical imaging, deep learning models trained on one dataset are often hard to generalize to other related datasets. Generally, biomedical images can be represented as points on a high-dimensional non-linear manifold. Failure of segmentation and classification algorithms to generalize across imaging modalities, patients, image acquisition protocols, medical centers, healthy and diseased populations, age, etc., can be explained by significant differences in the statistical distributions of datasets on the image manifolds, known as *covariate shift* [2]. Addressing covariate shift by retraining deep learning models on each new data

domain is impractical in most applications because of the scarcity of expert labeled data. Therefore, it is important to develop deep learning methods that generalize well to new related datasets not seen during training using few or no annotated examples from the new dataset. *Domain adaptation* [3] and *domain generalization* [4] paradigms aim at reducing the covariate shift between the training and test distributions by learning domain invariant features. Domain adaptation learns a feature representation that is invariant to the statistics of the source and target domains, and is discriminative enough for the actual learning task. Domain adaptation could either be unsupervised or semi-supervised. Domain generalization, a relatively less studied and harder problem, trains models using a variety of source domains to learn a generic feature representation which should perform well on unseen target domains. This flavor of transfer learning does not use any samples from the target distribution during training. Relatedly, few-shot learning is a paradigm which adapts a trained model to a completely new data distribution with very limited labeled training examples [5].

The *biomedical imaging* community has witnessed several applications of domain adaptation and few-shot learning. Adaptation across different medical centers is a known challenge in image segmentation [6], and has been achieved through both unsupervised [7], and supervised approaches [8]. Cross-modality domain adaptation methods between magnetic resonance (MR) and CT images have been proposed using variational autoencoders [9] for whole heart segmentation, and CycleGANs for segmentation of the prostate [10], [11]. Decision forests have been employed to adapt between in-vivo and in-vitro images [12] for intravascular ultrasound tissue segmentation. A few-shot network [13] was proposed to segment multiple organs in MR images. However, a priori knowledge of the unseen test domain is not always available, which hinders model generalizability as discussed above.

Very recently, some groups explored domain generalization for biomedical imaging. (1) A series of nine data augmentation techniques were applied to the training domains to mimic the test distribution [14] for heart ultrasound, heart and prostate MR image segmentation. (2) An episodic training-based meta-learning method [15] was applied to segment brain tissue in T1-weighted MRI across four medical centers. (3) A variational auto-encoder [16] was used to learn three latent subspaces to generalize across patients for a 2D cell segmentation task via domain disentanglement. These methods have certain limitations respectively: (1) In [14], the training data is very large and heavily augmented, which might not be the general case for many problems in medical imaging where the goal is to extract enough information from very limited data for domain generalization; the method is not tested on diseased populations which could vary significantly in anatomies and shapes from a generic healthy population. (2) In [15], again, the training set contains similar anatomies in both the train and test sets; the average performance is quite marginal than the compared baseline, with an improvement of only around 0.8% in Dice score (Baseline: 90.6% vs proposed: 91.4%); does not evaluate on cases with atrophied or irregular brain anatomies. (3) In [16], the method uses domain labels as an additional cue, which

we argue is difficult to define precisely, due to its wide range of interpretation; the average performance is not significant with an improvement of 0.4% in average Dice score (Baseline: 95.4% vs proposed: 95.8%); the method evaluates only one 2D dataset, not extending to the 3D case; each patient is considered as different domain acquired at the same medical center, and hence the test and train sets might have similar statistical distributions.

**Contributions.** In the present work, we extend a gradient-based meta-learning domain generalization method (MLDG) [1] that has shown promise for image classification tasks to the context of biomedical image segmentation, termed **MLDG-Seg**. To evaluate this approach, we focus on the problem of vertebrae segmentation in CT images and utilize three publicly available databases. To address the above-mentioned drawbacks of existing domain generalization methods, we construct three domain generalization contexts: (a) *generalization to new anatomies*: the vertebrae are divided into four domains: lumbar, lower, middle, and upper thoracic regions. The model is trained on three domains, and then tested on the unseen fourth domain. (b) *generalization to a diseased population with fractured vertebrae, dislocated discs*: the model is trained on a healthy population, then tested on unseen data of a diseased population from *another medical center*. (c) *generalization to unseen anatomies, surgical implants, different acquisition protocols, arbitrary orientations, and field of view (FoV)*: the model is tested on a very large dataset. Through these three contexts, we show that MLDG-Seg is able to learn generalized representations from very *limited training examples*. Finally, we show that the learned generalized representation can be quickly adapted with a few examples from the unseen target distribution in a  $k$ -shot learning setting to achieve additional performance gains.

## 2 Methodology

Meta-learning, or *learning to learn* [17], aims at learning a variety of tasks, and then quickly adapting to new tasks in different settings. We adopt the optimization-based model-agnostic method proposed in [1], called Meta Learning Domain Generalization (MLDG). Here, we briefly describe the method.

**Description.** Let there be two distributions: source  $\mathcal{S}$ , and target  $\mathcal{T}$ . Both  $\mathcal{S}$ , and  $\mathcal{T}$  share the same task, for example, segmentation or classification with the same label space. The goal of MLDG is to learn a single set of model parameters  $\theta$  via gradient descent and two meta-learners: meta-train and meta-test procedures. The model is trained on only the source  $\mathcal{S}$  domains, and then tested on target  $\mathcal{T}$  domain. The source domains  $\mathcal{S}$  are split into two sets: meta-train domains  $\hat{\mathcal{S}}$ , and meta-test domains  $\tilde{\mathcal{S}} = \mathcal{S} - \hat{\mathcal{S}}$ . The goal of the two splits is to mimic the setting of domain shifts, and thereby make it easier for the model to generalize on an unseen target domain  $\mathcal{T}$ . We reproduce the learning procedure in Algorithm 1, and show the extended version in Fig. 1.

**Explanation.** Let's consider a motivating example (carried out later in Experiment 1) from the image manifold of lumbar ( $\mathcal{D1}$ ), lower thoracic ( $\mathcal{D2}$ ) and middle thoracic regions ( $\mathcal{D3}$ ), comprising the set of source domains  $\mathcal{S}$ ; as well as

**Algorithm 1** MLDG

---

```

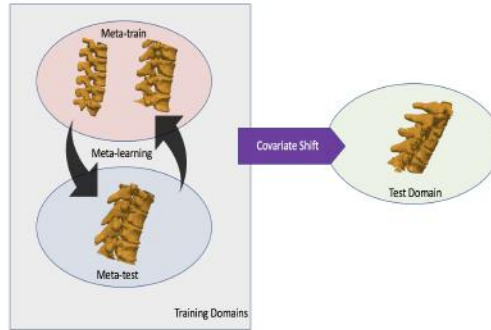
1: Input: Source domains  $\mathcal{S}$ 
2: Model parameters  $\theta$  and Hyperparameters:  $\alpha, \beta, \gamma$ .
3: for  $iterations = 1, 2, \dots$  do
4:   Randomly Split source domains  $\mathcal{S}$  into meta-train  $\hat{\mathcal{S}}$ , and meta-test  $\bar{\mathcal{S}}$ 
5:   Meta-train: Gradients  $\nabla_{\theta} = F'_{\theta}(\hat{\mathcal{S}}; \theta)$ 
6:   Updated parameters:  $\theta' \leftarrow \theta - \alpha \nabla_{\theta}$ 
7:   Meta-test: Compute Loss with updated parameter  $\theta'$  as  $G(\bar{\mathcal{S}}; \theta')$ 
8:   Final Model parameters:  $\theta \leftarrow \theta - \gamma \frac{\partial (F(\hat{\mathcal{S}}; \theta) + \beta G(\bar{\mathcal{S}}; \theta - \alpha \nabla_{\theta}))}{\partial \theta}$ 
9: end for

```

---

upper ( $\mathcal{D}_4$ ) thoracic region, which constitutes the unseen test  $\mathcal{T}$  domain. Refer to Algorithm 1. The MLDG method is supposed to learn a single model parameter  $\theta$  with the help of two optimization steps. At every iteration, the set of images in the source domains, here ( $\mathcal{D}_1$ ,  $\mathcal{D}_2$ , and  $\mathcal{D}_3$ ) are randomly split into a meta-train (for example, consisting of images from  $\mathcal{D}_1$ ,  $\mathcal{D}_2$  set), and meta-test with the set  $\mathcal{D}_3$ . Now, two losses are computed. The first loss  $\mathcal{F}$  is computed using the training examples from meta-train set and the gradient is computed with respect to the model parameter  $\theta$ . The second loss  $\mathcal{G}$  is computed on the meta-test set with the updated parameter  $\theta' = \theta - \alpha \nabla_{\theta}$ . The key idea by introducing the second loss in the meta-test stage is that an improvement of the model’s performance on the meta-train set should also improve the model’s performance on the meta-test set. The final model parameter  $\theta$  is updated by taking the gradient of the weighted combination of the two losses  $\mathcal{F}$ , and  $\mathcal{G}$ . By doing so, the model is tuned in such a way that performance is improved in both meta-train, and meta-test domains. In other words, the model is regularized and does not overfit to one particular domain, by finding the best possible gradient direction due to the joint optimization of the two losses. Compare this to a “vanilla” setup, where a model is directly given images from the three domains  $\mathcal{D}_1$ ,  $\mathcal{D}_2$ , and  $\mathcal{D}_3$  without any meta-learning setup, might overfit to a domain by minimizing its loss, and maximizing the loss for the other domains.

**Fig. 1. MLDG-Seg:** A schematic. Our method extends the MLDG algorithm to a segmentation by using a 3D Unet-like architecture as the backbone. To explain the segmentation procedure, consider the four domains: lumbar ( $\mathcal{D}_1$ ), lower ( $\mathcal{D}_2$ ), middle ( $\mathcal{D}_3$ ), upper ( $\mathcal{D}_4$ ) thoracic regions. As an example, the domains  $\mathcal{D}_1$ , and  $\mathcal{D}_2$  comprise the meta-train set, and  $\mathcal{D}_3$  the meta-test. The domain  $\mathcal{D}_4$  is the held-out test domain.



### 3 Experiments

#### 3.1 Databases

We validate our approach on three publicly available CT vertebrae segmentation datasets. See supplementary material for sample images of the three datasets.

**CSI challenge - healthy cases.** The datasets of spine CT (MICCAI 2014 challenge [18]) were acquired during daily clinical routine work in a trauma center from 10 adults (age: 16 to 35 years). In each subject, all 12 thoracic and 5 lumbar vertebrae were manually segmented to obtain ground truth.

**xVertSeg segmentation challenge - pathology cases.** This database consists of fractured and non-fractured CT lumbar spine images. We used the 15 subjects, ranging in age from 40 to 90 years, made publicly available with their corresponding lumbar ground truth segmentations [19].

**VerSe MICCAI segmentation challenge 2020 - versatile dataset.** This database consists of labeled lumbar, thoracic, cervical vertebrae across 100 cases in the released set [20], [21] as of June 2020. The data comes from several medical centers, with a very wide range of acquisition protocols, certain vertebrae with surgical implants, and a range of FoV with arbitrary image orientations.

#### 3.2 Experimental setup

We define the different implemented procedures, and then perform three experiments. **Baseline** is the vanilla 3D Unet-like [23] architecture. The model is trained on the source domains, and then tested on the unseen held-out test domain. The procedure is then repeated for the other domains. **MLDG-Seg** is the vanilla 3D Unet-like architecture trained under the scheme in Algorithm 1. The model is trained on the source domains, but at every iteration the source domains are divided into meta-train and meta-test splits. The model is then tested on unseen held-out test domain. The procedure is then repeated for the other domains.  **$k$ -shot learning** Once the MLDG-Seg model has been trained on source domains, it can be quickly adapted via fine-tuning using very limited labeled examples from the unseen target domain. The weights of the encoder, and bottleneck layers of the 3D Unet-like architecture were frozen in the  $k$ -shot learning experiments for fine-tuning. Our approach of  $k$ -shot learning is different than that of *test time adaptation* methods [24–27], where usually the encoder weights are not frozen in a Y-shaped network architecture, but are explicitly updated using a few examples at the time of testing. In contrast, we fine-tune the decoder weights with  $k$  examples from the test distribution. **Oracle** To establish a theoretical upper bound on segmentation accuracy, we train the vanilla 3D Unet-like architecture using labeled examples from the target domain, and then test on held-out examples from the target domain. This is presumably the easiest task, since the training and test domain distribution are similar. However, in our relatively small datasets the amount of available data for the oracle experiment is smaller than for the domain generalization experiments, so oracle performance reported below should be read with caution.

**Experiment 1.** The images in all the subjects of the CSI database are divided into different regions: lumbar (L1-L5), lower thoracic (T9-T12), middle thoracic (T5-T8), and upper thoracic vertebrae (T1-T4). Each of these regions comprise of a **domain**. Here, the aim is to let the model generalize across the underlying vertebral anatomy, the intensity profile and the surrounding inter-vertebral disc space which varies significantly along the vertebral column.

**Experiment 2.** Here, the four domains i.e., lumbar, lower, middle, and upper thoracic vertebrae from the CSI healthy database comprise the set of source domains. The xVertSeg database with pathology cases comprise the unseen target domain. Here, the aim is to let the model generalize from the vertebrae of healthy subjects to the structure of fractured or dislocated vertebrae images obtained at a different medical center.

**Experiment 3.** Here, the model is trained on CSI and xVertSeg datasets and tested on VerSe dataset, with the aim to generalize to unseen anatomies, different acquisition protocols, arbitrary orientations, FoVs, and pathology.

**Train, Validation, and Test data split-up details.** The supplementary material details the training, validation, and test sets for all three databases.

**Implementation details.** All the images were converted to 1 mm<sup>3</sup> isotropic resolution using FreeSurfer [22]. The images were standardized, using mean subtraction and division by standard deviation, and then normalized between 0 and 1. We implemented a 3D Unet-like [23] architecture in PyTorch [28]. This 3D Unet is the backbone architecture used for all the experiments. Each of the three hyperparameters  $\alpha$ ,  $\beta$ , and  $\gamma$  in MLDG (Algorithm 1) are set to 1. We use stochastic gradient descent as the optimizer with a learning rate of 0.001, momentum of 0.9, and a weight decay of  $5 \times 10^{-5}$ , dropout with probability of 0.3, and *groupnorm* as the normalization technique. See supplementary material for more details on the network architecture. The loss function used is Generalized Dice Loss [29]. We randomly sample 50 patches of size 64x64x64 from each subject in a domain, and perform data augmentation by randomly rotating and flipping 30 patches out of these 50 patches for every procedure. For the  $k$ -shot procedure, we randomly sampled 5 patches from each of the  $k$ -th subject from the unseen distribution. We trained every procedure, as described above, in experiment 1 for 10 epochs, and experiments 2 and 3 for 15 epochs. We use the model that gave the highest Dice score on the validation set to evaluate the unseen test domain. The models were trained on Nvidia P100 Tesla GPUs. A sliding window approach was used to obtain the predicted segmentations, which were then post-processed by retaining the largest connected component.

**Evaluation details.** We compute Dice coefficient (%), a volume-based measurement, and Average Symmetric Surface Distance (ASSD) in mm, a surface-based metric between the groundtruth image and a given model procedure segmentation output. *Desired: Higher Dice, and lower ASSD scores.* Furthermore, we perform pairwise Wilcoxon signed-rank test [30] between the baseline and each of the other procedures in all the three experiments for both Dice score and ASSD. In the three Tables 1, 2 and 3, we highlight the procedures which

reach significance at an  $\alpha=0.05$  significance value using the following notation to denote the level of significance: \* ( $p < 0.05$ ), \*\* ( $p < 0.005$ ), and \*\*\* ( $p < 0.0005$ ).

## 4 Results and Discussion

**Experiment 1.** Table 1 tabulates the results on different held-out test distributions. Each model is trained on three domains and then tested on the fourth unseen domain, except the oracle which is trained and tested on the same domain. MLDG-Seg consistently outperforms the baseline on both Dice and ASSD, and shows the desired low variance amongst the subjects. With a very few labeled examples from the test distribution, we see a further boost in performance. Fig. 2C shows that the MLDG-Seg is better able to segment the region of interest (ROI) by having significantly reduced background error than the baseline, in addition to the correct delineation of intervertebral discs (IVDs) when the held-out distribution is the middle thoracic region. A further improvement in performance is obtained using additional  $k$ -shot examples over MLDG-Seg. Ideally, the oracle should have the best performance than the rest of the procedures as the test domain distribution is similar to the training domain distribution. Here in this particular experimental setup, it is *not* surprising to see that the MLDG-Seg (and in some test domains, the baseline) performs better than the oracle. This might be due to the limited number of training (and validation) examples available to train the oracle. Furthermore, since all the procedures were trained for a fixed number of epochs, the oracle might have had a disadvantage of being trained for a lesser number of gradient steps than the MLDG-Seg procedure. An alternative approach would be to train a single oracle model which learns from all the domains simultaneously, and then test on each domain separately.

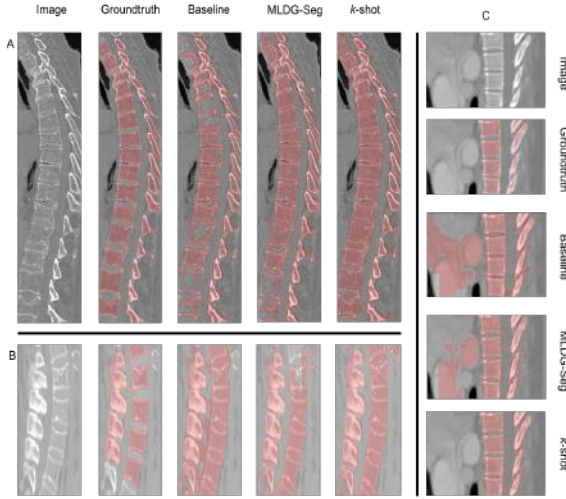
**Experiment 2.** Table 2 and the Spaghetti plots in Fig. 3 shows that the MLDG-Seg outperforms the baseline. Fig. 2B shows that MLDG-Seg is able to delineate the vertebrae by *not* segmenting the undesired spinal cord, as incorrectly segmented by the baseline. The  $k$ -shot setting further improves the segmentation. Here again, it is *not* necessarily surprising to see that MLDG-Seg performs better than the oracle. The oracle was trained with a smaller number of subjects than the baseline and MDG-Seg (see supplement).

**Experiment 3.** Table 3 shows the improved performance of MLDG-Seg over the baseline, and consistent performance boost in the few-shot learning regime. The  $k$ -shot procedure is either on par with the oracle or outperforms the same for  $k=4, 5$  and  $6$ . Fig. 2A depicts the superior performance of MLDG-Seg over the baseline on a compression fraction subject from a different distribution than the training set, where MLDG-Seg is able to segment more vertebrae completely than the baseline. The  $k$ -shot setting further improves the performance by segmenting the remaining vertebrae not segmented by baseline or MLDG-Seg. Here, the oracle performs better than most of the procedures, which is due to the fact that the training set for oracle is relatively larger than in experiments 1 and 2.

**Table 1.** Dice score (%), and ASSD in mm (mean  $\pm$  std. dev) for Experiment 1. Each row reports the result on held-out unseen test domain, where the model was trained on the remaining three domains. Supplement contains results on additional subjects.

Test domain	Baseline	MLDG-Seg	$k=1$	$k=2$	Oracle
Lumbar	81.67 $\pm$ 8.45 (1.85 $\pm$ 0.75)	87.85 $\pm$ 2.73 (1.42 $\pm$ 0.31)	88.57 $\pm$ 1.53 (1.46 $\pm$ 0.24)	88.19 $\pm$ 2.47 (1.69 $\pm$ 0.67)	83.60 $\pm$ 2.68 (2.74 $\pm$ 0.37)
Lower Th	83.52 $\pm$ 4.12 (2.66 $\pm$ 0.98)	86.17 $\pm$ 2.17 (1.44 $\pm$ 0.09)	81.44 $\pm$ 2.46 (2.73 $\pm$ 0.49)	82.28 $\pm$ 1.49 (2.74 $\pm$ 0.55)	80.25 $\pm$ 4.16 (2.32 $\pm$ 0.41)
Middle Th	55.72 $\pm$ 4.13 (10.41 $\pm$ 0.42)	64.36 $\pm$ 11.45 (6.85 $\pm$ 3.18)	75.57 $\pm$ 5.60 (3.56 $\pm$ 1.64)	76.98 $\pm$ 8.66 (2.20 $\pm$ 1.45)	83.60 $\pm$ 0.58 (2.21 $\pm$ 0.54)
Upper Th	82.00 $\pm$ 1.45 (1.68 $\pm$ 0.18)	83.70 $\pm$ 2.19 (1.46 $\pm$ 0.20)	74.47 $\pm$ 6.21 (4.83 $\pm$ 2.56)	81.50 $\pm$ 4.07 (1.75 $\pm$ 0.39)	75.84 $\pm$ 4.00 (1.93 $\pm$ 0.36)

**Fig. 2.** Qualitative illustrations for the three experiments. (A) Experiment 3: tested on VerSe dataset, here:  $k=7$ . (B) Experiment 2: tested on xVert dataset, here:  $k=4$ . (C) Experiment 1: shown is the middle thoracic region as the test set, here:  $k=2$ . Qualitative results for lumbar, lower, and upper thoracic regions can be found in the supplement. A minor discrepancy at the boundaries is noticed in groundtruth in C, perhaps due to registration errors.

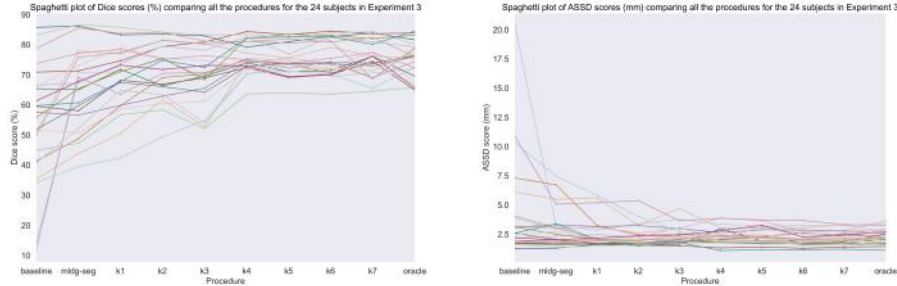


**Table 2.** Dice score (%), and ASSD in mm (mean  $\pm$  std. dev) for Experiment 2. The model was trained on CSI dataset and tested on xVertSeg dataset, thereby generalizing to a pathology dataset when trained on a healthy population.

Baseline	MLDG-Seg	$k=1$	$k=2$	$k=3$	$k=4$	Oracle
74.13 $\pm$ 13.69 (3.16 $\pm$ 1.05)	75.34 $\pm$ 12.10 (2.42 $\pm$ 0.70**)	76.88 $\pm$ 7.35 (2.74 $\pm$ 0.94*)	76.51 $\pm$ 8.85 (2.31 $\pm$ 0.65**)	77.95 $\pm$ 8.79* (2.38 $\pm$ 0.74**)	79.20 $\pm$ 8.01* (2.29 $\pm$ 0.77**)	74.94 $\pm$ 7.84 (3.98 $\pm$ 1.62)

**Conclusion** In the present work, we benchmarked the performance of a gradient-based meta-learning domain generalization segmentation method in the context of biomedical image analysis, across a variety of training and test settings. The method was not only able to generalize across multiple medical sites and scanners, which is the most widely studied problem of generalization, but was also able to generalize to newly introduced settings of unseen complex vertebrae anatomies, surrounding inter-vertebral discs space, varying bone and soft tissue intensities distribution, diseased populations, different acquisition proto-





**Fig. 3.** Spaghetti plots are shown for Experiment 3. *Left:* Dice coefficient, *Right:* ASSD score. Y-axis shows the Dice (%), and the ASSD (mm) scores for the different procedures shown on the X-axis. Each of the plotted lines denote a subject. Therefore, one can track the performance of a procedure for a given subject. See supplementary material for the spaghetti plots for all the other experiments.

Procedure	Dice score	ASSD
Baseline	$54.58 \pm 18.16$	$4.23 \pm 4.21$
<b>MLDG-Seg</b>	<b><math>64.82 \pm 13.13^{***}</math></b>	<b><math>2.99 \pm 1.59^*</math></b>
$k=1$	$69.12 \pm 10.48^{***}$	$2.57 \pm 1.21^{**}$
$k=2$	$71.49 \pm 9.04^{***}$	$2.31 \pm 0.91^{***}$
$k=3$	$70.18 \pm 9.53^{***}$	$2.42 \pm 0.76^{**}$
$k=4$	$75.90 \pm 4.99^{***}$	$2.29 \pm 0.69^*$
$k=5$	$75.27 \pm 5.11^{***}$	$2.31 \pm 0.68^*$
$k=6$	$75.54 \pm 5.71^{***}$	$2.13 \pm 0.60^{**}$
$k=7$	$75.28 \pm 5.67^{***}$	$2.17 \pm 0.57^{**}$
Oracle	$74.97 \pm 5.81^{***}$	$2.31 \pm 0.62^*$

**Table 3.** Dice score (%), and ASSD in mm (mean  $\pm$  std. dev) for Experiment 3. The model was trained on CSI, and xVertSeg dataset and tested on VerSe dataset, thereby generalizing to fractured or dislocated vertebrae images obtained at different medical centers, unseen anatomies, different acquisition protocols, arbitrary orientations, FoVs, and various pathology.

cols, arbitrary orientations, and FoVs, thus resembling actual clinical settings. In future, we will evaluate the method on other modalities such as MRI, and US and compare with the recently proposed domain generalization methods. Our source code, scripts, and dataset-split files are available at:

<https://github.com/Pulkit-Khandelwal/medical-mldg-seg>.

**Acknowledgments** This work was supported by NIH grant R01 EB017255.

## References

1. Li, D., Yang, Y., Song, Y.Z. and Hospedales, T.M., 2018, April. Learning to generalize: Meta-learning for domain generalization. In Thirty-Second AAAI Conference on Artificial Intelligence.
2. Meng, Q., Rueckert, D., and Kainz, B. (2020). Learning Cross-domain Generalizable Features by Representation Disentanglement. arXiv preprint arXiv:2003.00321.
3. Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M. and Lempitsky, V., 2016. Domain-adversarial training of neural networks. The Journal of Machine Learning Research, 17(1), pp.2096-2030.

4. Li, D., Yang, Y., Song, Y.Z. and Hospedales, T.M., 2017. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision* (pp. 5542-5550).
5. Snell, J., Swersky, K. and Zemel, R., 2017. Prototypical networks for few-shot learning. In *Advances in neural information processing systems* (pp. 4077-4087).
6. Glocker, B., Robinson, R., Castro, D.C., Dou, Q. and Konukoglu, E., 2019. Machine learning with multi-site imaging data: An empirical study on the impact of scanner effects. *arXiv preprint arXiv:1910.04597*.
7. Kamnitsas, K., Baumgartner, C., Ledig, C., Newcombe, V., Simpson, J., Kane, A., Menon, D., Nori, A., Criminisi, A., Rueckert, D. and Glocker, B., 2017, June. Unsupervised domain adaptation in brain lesion segmentation with adversarial networks. In *International conference on information processing in medical imaging* (pp. 597-609). Springer, Cham.
8. Valindria, V.V., Lavdas, I., Bai, W., Kamnitsas, K., Aboagye, E.O., Rockall, A.G., Rueckert, D. and Glocker, B., 2018. Domain adaptation for MRI organ segmentation using reverse classification accuracy. *arXiv preprint arXiv:1806.00363*.
9. Ouyang, C., Kamnitsas, K., Biffi, C., Duan, J. and Rueckert, D., 2019, October. Data efficient unsupervised domain adaptation for Cross-Modality image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 669-677). Springer, Cham.
10. Liu, Y., Khosravan, N., Liu, Y., Stember, J., Shoag, J., Bagci, U. and Jambawalikar, S., 2019. Cross-Modality Knowledge Transfer for Prostate Segmentation from CT Scans. In *Domain Adaptation and Representation Transfer and Medical Image Learning with Less Labels and Imperfect Data* (pp. 63-71). Springer, Cham.
11. Dou, Q., Ouyang, C., Chen, C., Chen, H., Glocker, B., Zhuang, X. and Heng, P.A., 2019. Pnp-adanet: Plug-and-play adversarial domain adaptation network at unpaired cross-modality cardiac segmentation. *IEEE Access*, 7, pp.99065-99076.
12. Conjeti, S., Katouzian, A., Roy, A.G., Peter, L., Sheet, D., Carlier, S., Laine, A. and Navab, N., 2016. Supervised domain adaptation of decision forests: Transfer of models trained in vitro for in vivo intravascular ultrasound tissue characterization. *Medical image analysis*, 32, pp.1-17.
13. Roy, A.G., Siddiqui, S., Plsterl, S., Navab, N. and Wachinger, C., 2020. Squeeze and excite-guided few-shot segmentation of volumetric images. *Medical image analysis*, 59, p.101587.
14. Zhang, L., Wang, X., Yang, D., Sanford, T., Harmon, S., Turkbey, B., Wood, B.J., Roth, H., Myronenko, A., Xu, D. and Xu, Z., 2020. Generalizing deep learning for medical image segmentation to unseen domains via deep stacked transformation. *IEEE Transactions on Medical Imaging*.
15. Dou, Q., de Castro, D.C., Kamnitsas, K. and Glocker, B., 2019. Domain generalization via model-agnostic learning of semantic features. In *Advances in Neural Information Processing Systems* (pp. 6450-6461).
16. Ilse, M., Tomczak, J.M., Louizos, C. and Welling, M., 2019. Diva: Domain invariant variational autoencoders. *arXiv preprint arXiv:1905.10427*.
17. Finn, C., Abbeel, P. and Levine, S., 2017, August. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70* (pp. 1126-1135). JMLR. org.
18. Yao, J., Burns, J.E., Forsberg, D., Seitel, A., Rasouljan, A., Abolmaesumi, P., Hammernik, K., Urschler, M., Ibragimov, B., Korez, R. and Vrtovec, T., 2016. A multi-center milestone study of clinical vertebral CT segmentation. *Computerized Medical Imaging and Graphics*, 49, pp.16-28.

19. R. Korez, B. Ibragimov, B. Likar, F. Pernu and T. Vrtovec, "A Framework for Automated Spine and Vertebrae Interpolation-Based Detection and Model-Based Segmentation," in *IEEE Transactions on Medical Imaging*, vol. 34, no. 8, pp. 1649-1662, Aug. 2015.
20. Sekuboyina, A., Bayat, A., Husseini, M.E., Lffler, M., Rempfler, M., Kukaka, J., Tetteh, G., Valentinitich, A., Payer, C., Urschler, M. and Chen, M., 2020. VerSe: A Vertebrae Labelling and Segmentation Benchmark. *arXiv preprint arXiv:2001.09193*.
21. Lffler, M.T., Sekuboyina, A., Jacob, A., Grau, A.L., Scharr, A., El Husseini, M., Kallweit, M., Zimmer, C., Baum, T. and Kirschke, J.S., 2020. A Vertebral Segmentation Dataset with Fracture Grading. *Radiology: Artificial Intelligence*, 2(4), p.e190138.
22. Fischl, B., 2012. FreeSurfer. *Neuroimage*, 62(2), pp.774-781.
23. Ronneberger, O., Fischer, P. and Brox, T., 2015, October. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (pp. 234-241). Springer, Cham.
24. Wang, G., Li, W., Zuluaga, M.A., Pratt, R., Patel, P.A., Aertsen, M., Doel, T., David, A.L., Deprest, J., Ourselin, S. and Vercauteren, T., 2018. Interactive medical image segmentation using deep learning with image-specific fine tuning. *IEEE transactions on medical imaging*, 37(7), pp.1562-1573.
25. Sun, Y., Wang, X., Liu, Z., Miller, J., Efros, A.A. and Hardt, M., 2019. Test-time training for out-of-distribution generalization. *arXiv preprint arXiv:1909.13231*.
26. Karani, N., Chaitanya, K. and Konukoglu, E., 2020. Test-Time Adaptable Neural Networks for Robust Medical Image Segmentation. *arXiv preprint arXiv:2004.04668*.
27. Zhang, J., Liu, Z., Zhang, S., Zhang, H., Spincemaille, P., Nguyen, T.D., Sabuncu, M.R. and Wang, Y., 2020. Fidelity imposed network edit (FINE) for solving ill-posed image reconstruction. *NeuroImage*, 211, p.116579.
28. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L. and Desmaison, A., 2019. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems* (pp. 8024-8035).
29. Crum, W.R., Camara, O. and Hill, D.L., 2006. Generalized overlap measures for evaluation and validation in medical image analysis. *IEEE transactions on medical imaging*, 25(11), pp.1451-1461.
30. Wilcoxon, F., 1992. Individual comparisons by ranking methods. In *Breakthroughs in statistics* (pp. 196-202). Springer, New York, NY.

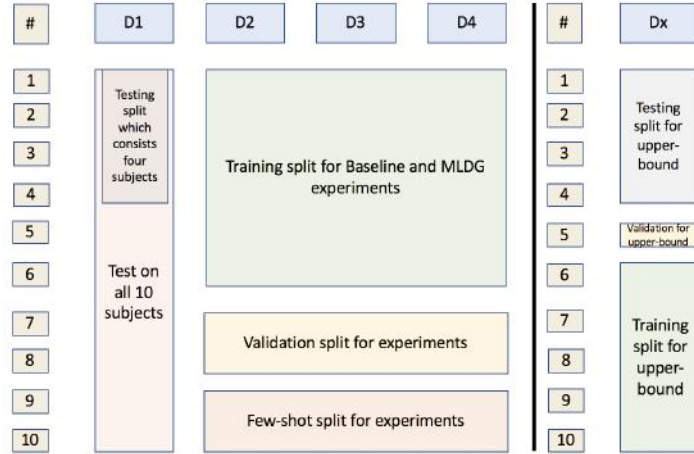
# Domain Generalizer: A Few-shot Meta Learning Framework for Domain Generalization in Medical Imaging [Supplementary Material]

Pulkit Khandelwal<sup>1,2</sup> ✉ and Paul Yushkevich<sup>2</sup>

<sup>1</sup> Department of Bioengineering, University of Pennsylvania, Philadelphia, PA, United States

<sup>2</sup> Penn Image Computing and Science Laboratory, Department of Radiology, University of Pennsylvania, Philadelphia, PA, United States  
pulks@seas.upenn.edu

## 1 Dataset split-up for train, validation, and test sets



**Fig. 1.** The CSI database consists of 10 subjects each comprising the four domains: lumbar, lower, middle, and upper thoracic vertebrae. The numerals in the above figure denotes the subject number. The data split-up on the left side of the black vertical bar gives the split up for the results in Table 1 of the main paper. The data split for the oracle (upper-bound) procedure in Table 1 in the main paper is depicted on the right side of the vertical bar. The results in Table 1 in the main paper is for the first four subjects. The results are then reported for all the four domains, leaving one out as test. The results on all the 10 subjects are given in Table 1 in Section 5 of this supplementary material. Note: Dx denotes any of the domains.

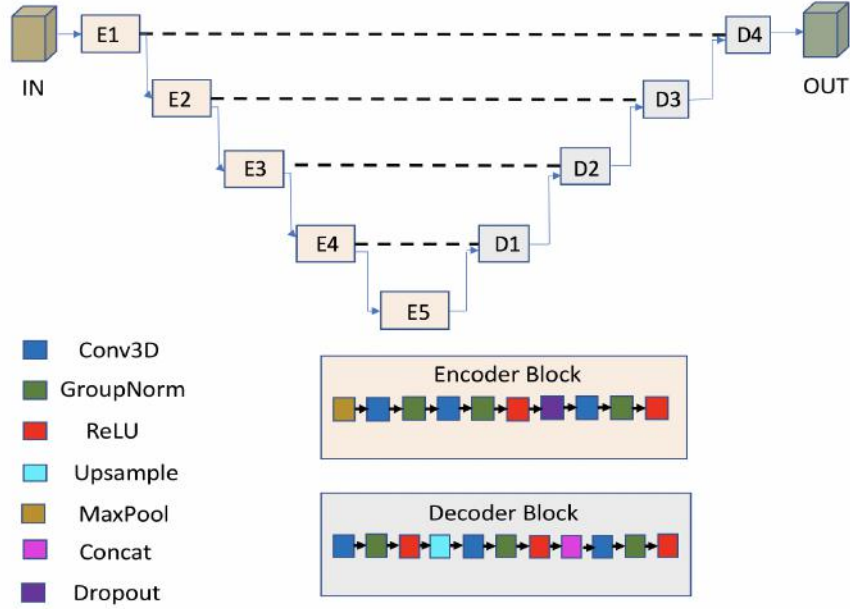


**Fig. 2.** The xVertSeg database consists of 15 subjects. Depicted above is the split for the results in Table 2 in the main paper. Note that oracle is the upper-bound in the figure.

Validation (11)	k-shot (7)	Oracle (20)	Test (24)
GL240_CT-ax verse004 verse127 verse257 verse405_verse212_CT-sag verse507 verse629_CT-iso verse807_CT-iso verse405_verse259_CT-sag verse403_verse255_CT-sag GL364_CT-ax	GL247 verse015 verse135 verse254 verse405_verse258_CT-sag verse619_CT-iso verse811_CT-ax	GL240_CT-ax verse004 verse127 verse257 verse405_verse212_CT-sag verse507 verse629_CT-iso verse807_CT-iso verse113 verse033 verse034 verse820_CT-iso verse407_verse215_CT-sag verse500 verse403_verse208_CT-sag verse406_verse261_CT-sag verse503 GL016 verse402_verse251_CT-sag verse631_CT-sag	verse112_CT-iso GL295_CT-ax verse406_verse214_CT-sag verse605_CT-sag verse111 verse005 GL090_CT-ax GL124_CT-ax verse641_CT-iso GL453_CT-ax verse122 GL003 GL047 verse808_CT-iso verse031 verse008 verse407_verse262_CT-sag verse514 verse402_verse202_CT-sag verse401_verse201_CT-iso verse504 verse818_CT-iso verse401_verse253_CT-iso verse104_CT-iso

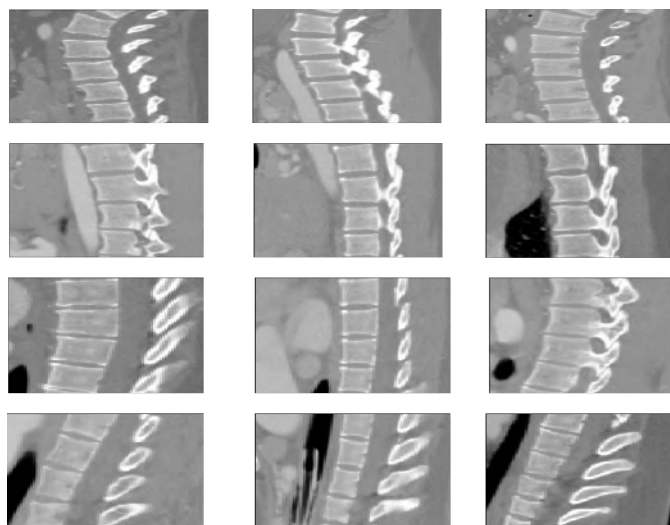
**Fig. 3.** The VerSe segmentation challenge dataset consists of 100 cases at the time of writing this manuscript in June 2020. We randomly select 55 cases, and assign them to the four categories as listed in the table. Shown are the subject IDs.

## 2 3D Unet-like architecture

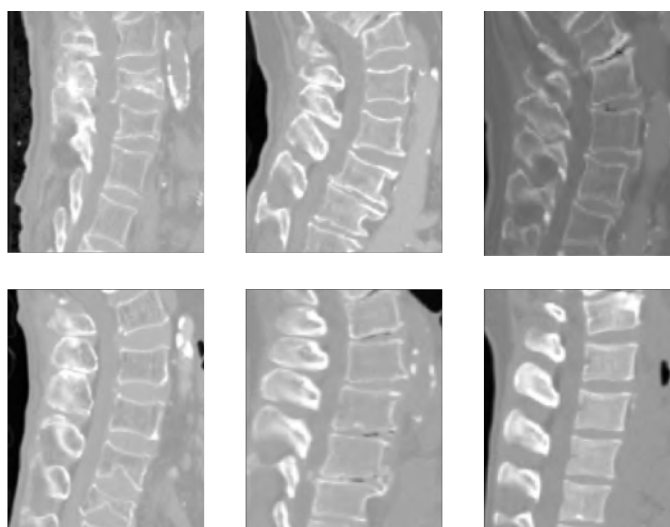


**Fig. 4.** 3D Unet-like architecture implemented. IN and OUT are the input and the output respectively each with dimensions  $1 \times 64 \times 64 \times 64$ , and  $2 \times 64 \times 64 \times 64$  (for the two classes). There are five encoder blocks (with the last block acting as the bottleneck layer). There are four decoder blocks. The dashed lines represent the skip connections. We have used dropout with probability of 0.3, and *groupnorm* as the normalization method. Note: the first encoder block does not consist of the MaxPool operation, and the first decoder block does not consist of the first set of conv3D, groupnorm, and ReLU units. The *encoder* block has the following *output* dimensions at each of the encoder block: E1:  $16 \times 64 \times 64 \times 64$ , E2:  $32 \times 32 \times 32 \times 32$ , E3:  $64 \times 16 \times 16 \times 16$ , E4:  $128 \times 8 \times 8 \times 8$ , E5:  $256 \times 4 \times 4 \times 4$ . The *decoder* block has the following *output* dimensions at each of the decoder block: D1:  $256 \times 8 \times 8 \times 8$ , D2:  $128 \times 16 \times 16 \times 16$ , D3:  $64 \times 32 \times 32 \times 32$ , D4:  $32 \times 64 \times 64 \times 64$ .

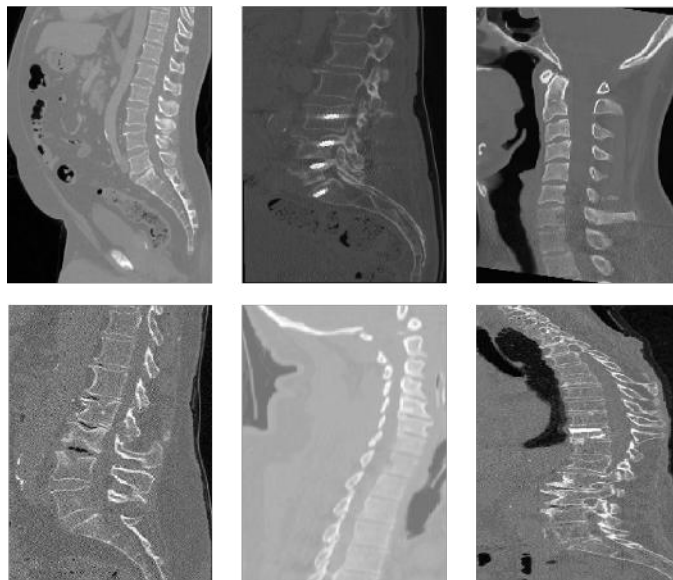
### 3 Sample image slices for the three datasets



**Fig. 5.** CSI dataset



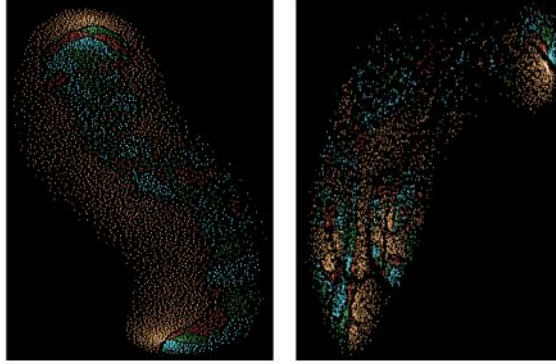
**Fig. 6.** xVertSeg dataset



**Fig. 7.** VerSe dataset



## 4 t-SNE plots



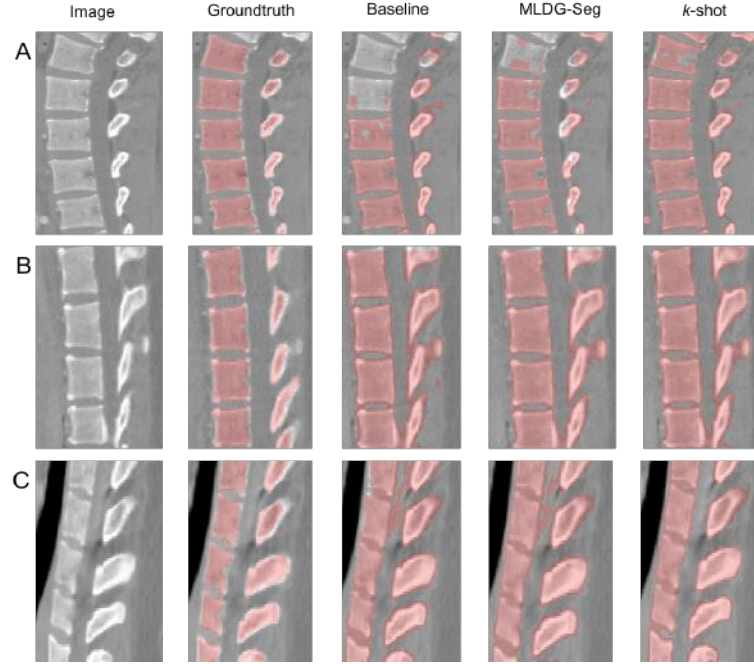
**Fig. 8.** Shown are the t-SNE [1] plots for Baseline (left), and MLDG-Seg (right). The trained bottleneck layer’s feature vector with dimension of 16384 (E5: 256x4x4x4) are plotted for the experiment when the model was trained on CSI and xVertSeg dataset (Experiment 3). The four colors represent the four domains: lumbar, lower thoracic, middle thoracic, and upper thoracic regions in the CSI dataset. It can be seen that the data is clustered more separately in the trained baseline than the MLDG-Seg model. **Whereas, the trained MLDG-Seg method learns domain agnostic features to better generalize the model.** The following hyperparameters were used to produce the t-sne plot: perplexity: 45, learning rate: 10, supervision: 30, iterations: 700.

## 5 Experiment 1 additional results

**Table 1.** Results for Experiment 1 on all the **10 subjects**. See Figure 1 for the data split-up. Dice score (%), and Average Symmetric Surface Distance (ASSD) in mm (mean  $\pm$  std. dev) for Experiment 1 within brackets. Each row reports the result on the held-out unseen test domain, where the model was trained on the remaining three domains. We performed the pairwise Wilcoxon signed-rank test between the baseline and MLDG-Seg for both Dice score and ASSD. We highlight the procedures which reach significance at an  $\alpha=0.05$  significance value using the following notation to denote the level of significance: \* ( $p<0.05$ ), \*\* ( $p<0.005$ ), and \*\*\* ( $p<0.0005$ ).

Test domain	Baseline	MLDG-Seg
Lumbar	81.22 $\pm$ 9.48 (1.90 $\pm$ 0.80)	88.31 $\pm$ 2.55* (1.39 $\pm$ 0.32*)
Lower Thoracic	83.58 $\pm$ 6.71 (2.39 $\pm$ 1.21)	86.77 $\pm$ 3.35 (1.41 $\pm$ 0.57*)
Middle Thoracic	65.5 $\pm$ 15.14 (7.49 $\pm$ 4.17)	71.22 $\pm$ 14.4 (4.93 $\pm$ 3.80**)
Upper Thoracic	81.55 $\pm$ 3.99 (1.59 $\pm$ 0.33)	81.75 $\pm$ 5.87 (1.70 $\pm$ 0.88)

## 6 Experiment 1 additional qualitative results

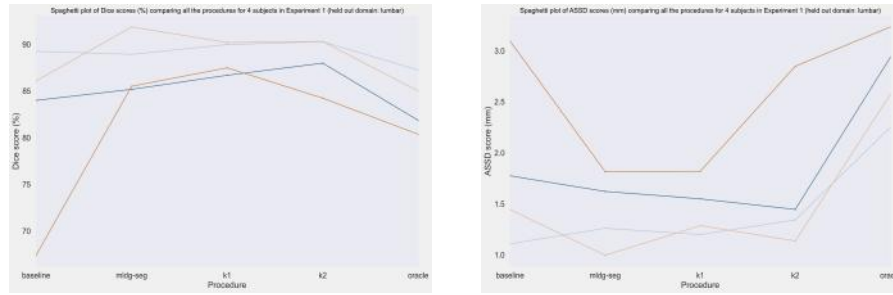


**Fig. 9.** Qualitative illustrations for the three held-out domains in experiment 1. (A) Held-out domain: Lumbar, here:  $k=2$ . (B) Held-out domain: Lower Thoracic, here:  $k=2$ . (C) Held-out domain: Middle Thoracic, here:  $k=2$ . A minor discrepancy at the boundaries is noticed in the groundtruth segmentations, perhaps due to registration errors when converting to an isotropic resolution.

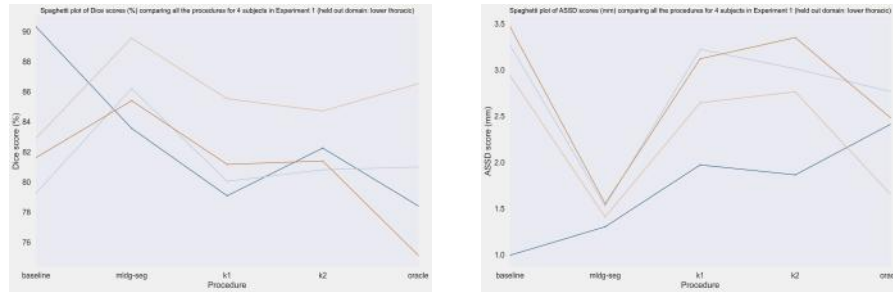
## 7 Spaghetti Plots

In the following set of figures, Spaghetti plots are shown for all the experiments. Y-axis shows the Dice (%), and the ASSD (mm) scores for the different procedures, shown on the X-axis. Each of the plotted lines denote a subject. Therefore, one can track the performance of a procedure for a given subject.

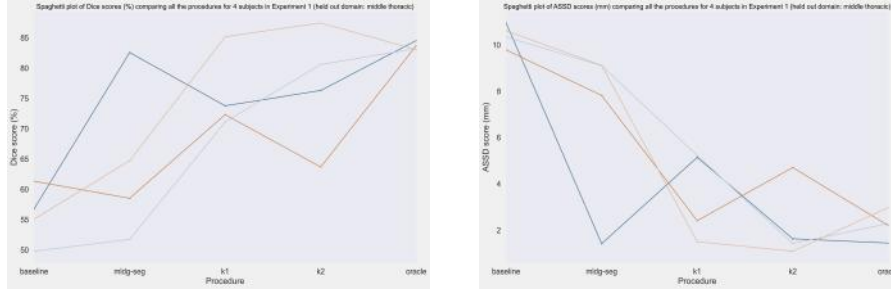
### 7.1 Experiment 1 (4 subjects) [Refer Table 1 in the main paper]



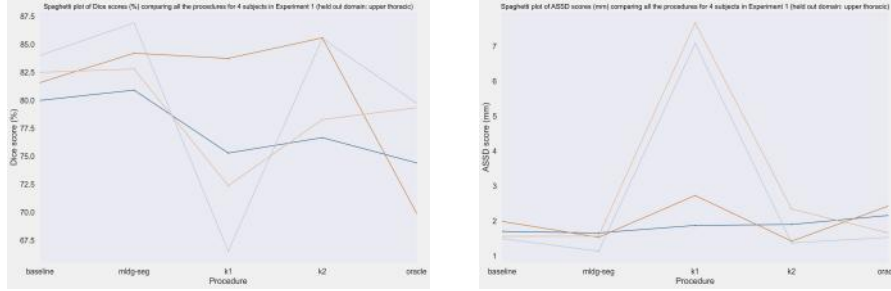
**Fig. 10.** *Left:* Dice coefficient, *Right:* ASSD score. Held-out domain: Lumbar.



**Fig. 11.** *Left:* Dice coefficient, *Right:* ASSD score. Held-out domain: Lower thoracic.

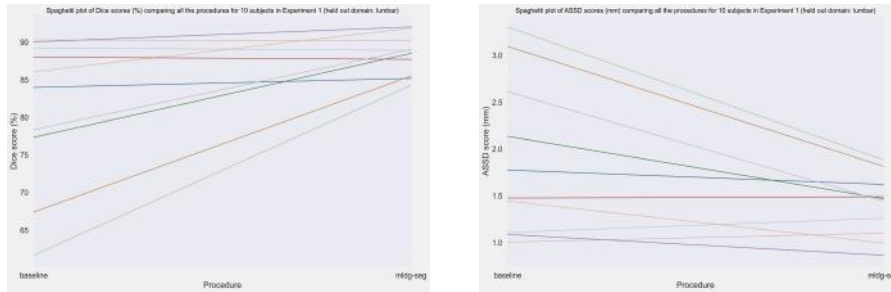


**Fig. 12.** *Left:* Dice coefficient, *Right:* ASSD score. Held-out domain: Middle thoracic.

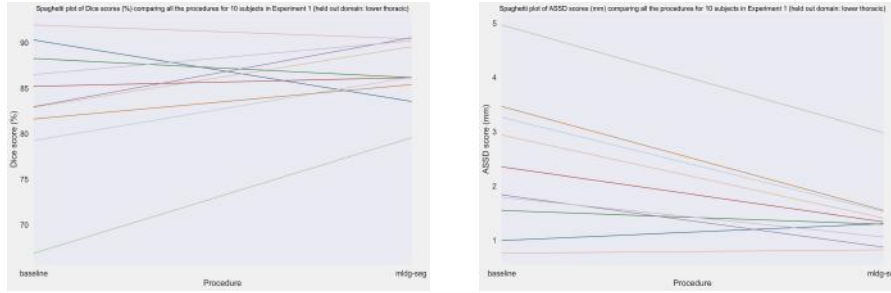


**Fig. 13.** *Left:* Dice coefficient, *Right:* ASSD score. Held-out domain: Upper thoracic.

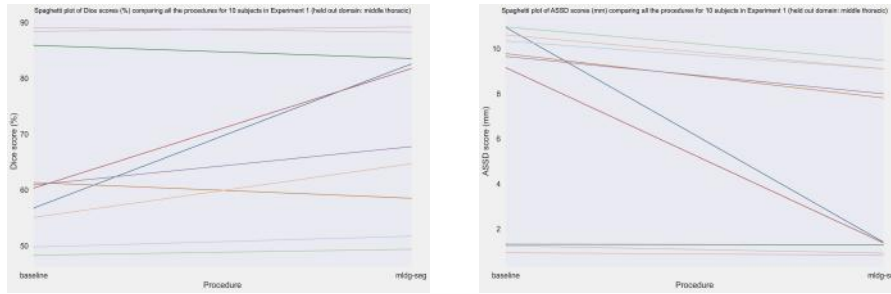
## 7.2 Experiment 1 (10 subjects) [Refer Table 1 in this supplement in Section 5 above]



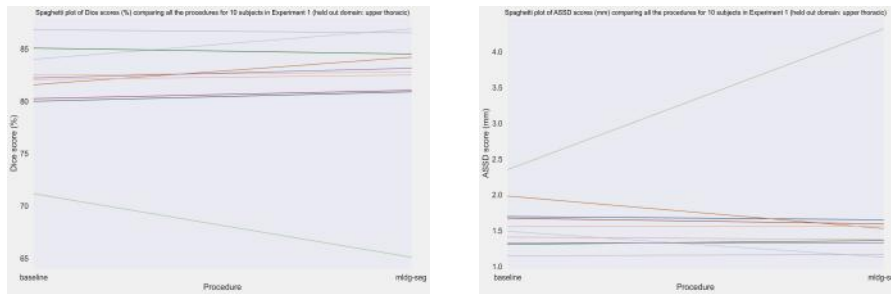
**Fig. 14.** *Left:* Dice coefficient, *Right:* ASSD score. Held-out domain: Lumbar.



**Fig. 15.** *Left:* Dice coefficient, *Right:* ASSD score. Held-out domain: Lower thoracic.

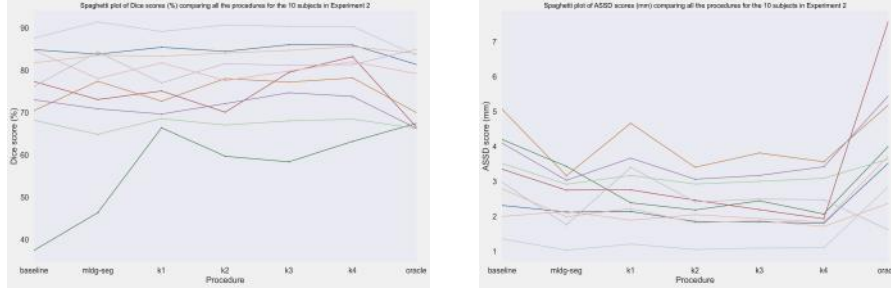


**Fig. 16.** *Left:* Dice coefficient, *Right:* ASSD score. Held-out domain: Middle thoracic.



**Fig. 17.** *Left:* Dice coefficient, *Right:* ASSD score. Held-out domain: Upper thoracic.

### 7.3 Experiment 2 [Refer Table 2 in the main paper]



**Fig. 18.** *Left:* Dice coefficient, *Right:* ASSD score. Experiment 2.

## References

1. Maaten, L.V.D. and Hinton, G., 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(Nov), pp.2579-2605.