



# A Lifelong Learning Approach to Brain MR Segmentation Across Scanners and Protocols

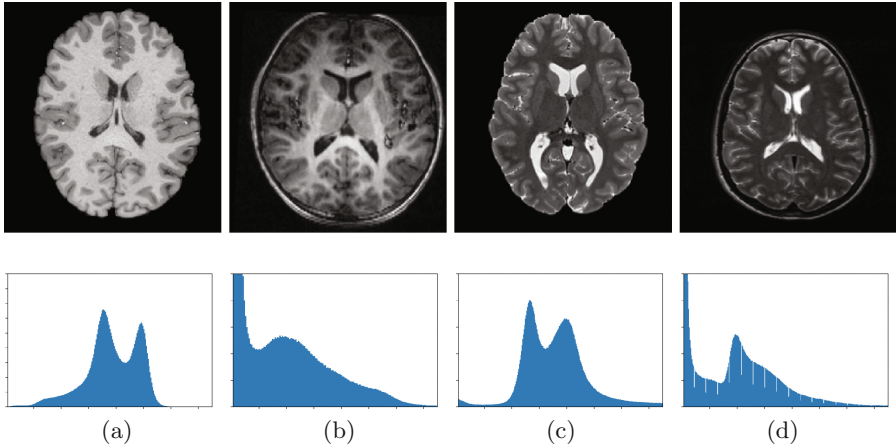
Neerav Karani<sup>(✉)</sup>, Krishna Chaitanya, Christian Baumgartner,  
and Ender Konukoglu

Computer Vision Lab, ETH Zurich, Zurich, Switzerland  
nkarani@vision.ee.ethz.ch

**Abstract.** Convolutional neural networks (CNNs) have shown promising results on several segmentation tasks in magnetic resonance (MR) images. However, the accuracy of CNNs may degrade severely when segmenting images acquired with different scanners and/or protocols as compared to the training data, thus limiting their practical utility. We address this shortcoming in a lifelong multi-domain learning setting by treating images acquired with different scanners or protocols as samples from different, but related domains. Our solution is a single CNN with shared convolutional filters and domain-specific batch normalization layers, which can be tuned to new domains with only a few ( $\approx 4$ ) labelled images. Importantly, this is achieved while retaining performance on the older domains whose training data may no longer be available. We evaluate the method for brain structure segmentation in MR images. Results demonstrate that the proposed method largely closes the gap to the benchmark, which is training a dedicated CNN for each scanner.

## 1 Introduction

Segmentation of brain MR images is a critical step in many diagnostic and surgical applications. Accordingly, several approaches have been proposed for tackling this problem such as atlas-based segmentation [1], methods based on machine learning techniques such as CNNs [2], among many others as detailed in this recent survey [3]. One of the important challenges in many MRI analysis tasks, including segmentation, is robustness to differences in statistical characteristics of image intensities. These differences might arise due to using different scanners in which factors like drift in scanner SNR over time [4], gradient non-linearities [5] and others play an important role. Intensity variations may even arise when scanning protocol parameters (flip angle, echo or repetition time, etc.) are slightly changed on the same scanner. Figure 1(a, b) shows 2D slices from two T1-weighted MRI datasets from different scanners, along with their intensity histograms which show the aforementioned variations. Segmentation algorithms are often very sensitive to such changes. Furthermore, images acquired with different MR modalities, such as T1 and T2-weighted images, may have considerable high-level similarities in image content (see Fig. 1). While analyzing these



**Fig. 1.** Image slices (top) and corresponding histograms (bottom) of normalized T1w (a,b) and T2w (c,d) MRIs from different scanners. Despite high-level information similarity, there exists considerable intensity and contrast differences, which segmentation algorithms are often sensitive to.

images, humans can leverage such commonalities easily and it would be highly desirable if learning-based algorithms could mimic this trait.

In the parlance of transfer machine learning, images acquired from different scanners, protocols or similar MR modalities may be viewed as data points sampled from different domains, with the degree of domain shift potentially indicated by the differences in their intensity statistics. This perspective motivates us to employ ideas from the literature of domain adaptation [6], multi-domain learning [7] and lifelong learning [8] to the problem of brain segmentation across scanners/protocols. Domain adaptation/transfer learning refers to a situation where a learner trained on a source domain is able to perform well on a target domain, of which only a few labelled examples are available. However, in this case, the performance on the source domain may not be necessarily maintained after adaptation. Multi-domain learning aims to train a learner that can simultaneously perform well on multiple domains. Finally, in lifelong learning, a multi-domain learner is able to incorporate new domains with only few labelled examples, while preserving performance on previous domains.

Variants of image intensity standardization [9, 10] and atlas intensity renormalization [11] have been proposed as pre-processing steps to insure conventional segmentation methods from inter-scanner differences. Among learning methods based on hand-crafted features, transfer learning approaches have been employed for multi-site segmentation [12] and classification [13]. While adaptive support vector machines used by [12] may be adapted for new scanners in a lifelong learning sense, they are likely to be limited by the quality of the hand-crafted features. Using CNNs, [14] propose to deal with inter-protocol differences by learning domain invariant representations. This approach may be limited to

work with the least common denominator between the domains, while, as shown in [15], providing a few separate parameters for each domain allows for learning of domain specific nuances. Further, it is unclear how [14] can be extended to deal with new domains that may be encountered after the initial training. In the computer vision literature, several adaptations of batch normalization (BN) [16] have been suggested for domain adaptation [17, 18] and multi-domain learning [15, 19] for object recognition using CNNs. Broadly, these works employ BN for domain-specific scaling to account for domain shifts, while sharing the bulk of the CNN parameters to leverage the similarity between the domains.

In this work, we extend approaches based on adaptive BN layers for segmentation across scanning protocols in a lifelong learning setting. In particular, we train a CNN with common convolutional filters and specific BN parameters for each protocol/scanner. The network is initially trained with images from a few scanners to learn appropriate convolutional filters. It can then be adapted to new protocols/scanners by fine-tuning the BN parameters with only a few labelled images. Crucially, this is achieved without performance degradation on the older scanners, whose training data is not available after the initial training.

## 2 Method

Batch normalization (BN) was introduced in [16] to enable faster training of deep neural networks by preventing saturated gradients via normalization of inputs before each non-linear activation layer. In a BN layer, each batch  $x_B$  is normalized as shown in Eq. 1. During training,  $\mu_B$  and  $\sigma_B^2$  are the mean and variance of  $x_B$ , while at test time, they are the estimated population mean and variance as approximated by a moving average over training batches.  $\gamma$ ,  $\beta$  are learnable parameters that allow the network to undo the normalization, if required. Inspired by [15], we propose to use separate batch normalization for each protocol/scanner.

$$BN(x_B) = \gamma \times \frac{x_B - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} + \beta \quad (1)$$

Notwithstanding variations in image statistics due to inter-scanner differences, a segmentation network would be confronted with images of the same organ, acquired with the same modality (MR). Hence, it is reasonable to postulate common characteristics between the domains and thus, shared support in an appropriate representation space. Following [15], we hypothesize that such a representation space can be found by using domain-agnostic convolutional filters and that the inter-domain differences can be handled by appropriate normalization via domain-specific BN modules. This approach is not only in line with the previous domain adaptation works [18], but also embodies the normalization idea of conventional proposals for dealing with inter-scanner variations [9–11]. Further, like [19], once suitable shared convolutional filters have been learned, we adapt the domain-specific BN layers to new related domains.

The training procedure in our framework is as follows. We use superscript  $^{bn}$  to indicate a network with domain-specific BN layers. We initially train a network,  $N_{12\dots d}^{bn}$  on  $d$  domains, with shared convolutional filters and separate BN parameters,  $bn_k$ , for each domain  $D_k$ . During training, each batch consists of only one domain, with all domains covered successively. In a training iteration when the batch consists of domain  $D_k$ ,  $bn_{k'}$  for  $k' \neq k$  are frozen. Now, consider a new domain  $D_{d+1}$ , with a few labelled images  $I_{D_{d+1}}$ . We split this small dataset into two halves, using one for training,  $I_{D_{d+1}}^{tr}$  and the other for validation,  $I_{D_{d+1}}^{vl}$ . We evaluate the performance of  $N_{12\dots d}^{bn}$  on  $I_{D_{d+1}}^{tr}$ , using each learned  $bn_k$ ,  $k = 1, 2, \dots, d$ . If  $bn_{k^*}$  leads to the best accuracy, we infer that among the already learned domains,  $D_{k^*}$  is the closest to  $D_{d+1}$ . Then, keeping the convolutional filter weights fixed, an additional set of BN parameters  $bn_{d+1}$  is initialized with  $bn_{k^*}$  and fine-tuned using  $I_{D_{d+1}}^{tr}$  with standard stochastic gradient descent minimization. The optimization is stopped when the performance on  $I_{D_{d+1}}^{vl}$  stops improving. Now, the network can segment all domains  $D_k$ , for  $k = 1, 2, \dots, d, d + 1$  using their respective  $bn_k$ .

In the spirit of lifelong learning, this approach allows learning on new domains with only a few labelled examples. This is enabled by utilizing the knowledge obtained from learning on the old domains, in the form of the trained domain-agnostic parameters. The fact that the number of domain-specific parameters is small comes with two advantages. One, that they can be tuned for a new domain by training with a few labelled images quickly and with minimal risk of overfitting. Secondly, they can be saved for each domain without significant memory footprint. Finally, catastrophic forgetting [20] by performance degradation on previous domains does not arise in this approach by construction because of the explicit separate modeling of shared and private parameters.

### 3 Experiments and Results

**Datasets:** Brain MR datasets from several scanners, hospitals, or acquisition protocols are required to test the applicability of the proposed method for lifelong multi-domain learning. To the best of our knowledge, there are only a few publicly available brain MRI datasets with ground truth segmentation labels from human experts. Therefore, we use FreeSurfer [1] to generate pseudo ground truth annotations. While annotations from human experts would be ideal, we believe that FreeSurfer annotations can serve as a reasonable proxy to test our approach to lifelong multi-scanner learning.

We use images from 4 publicly available datasets: Human Connectome Project (HCP) [21], Alzheimer’s Disease Neuroimaging Initiative (ADNI)<sup>1</sup>, Autism Brain Imaging Data Exchange (ABIDE) [22] and Information eXtraction from Images (IXI)<sup>2</sup>. The datasets are split into different domains, as shown in Table 1. Domains  $D_1, D_2, D_3$  are treated as initially available, and  $D_4, D_5$  as

<sup>1</sup> [adni.loni.usc.edu](http://adni.loni.usc.edu).

<sup>2</sup> [brain-development.org/ixi-dataset/](http://brain-development.org/ixi-dataset/).

**Table 1.** Details of the datasets used for our experiments.

Domain	Dataset	Field	MR Modality	$n_{train}$	$n_{train}^{scratch}$	$n_{test}$
D <sub>1</sub>	HCP	3T	T1w	30	30	20
D <sub>2</sub>	HCP	3T	T2w	30	30	20
D <sub>3</sub>	ADNI	1.5T	T1w	30	30	20
D <sub>4</sub>	ABIDE, Caltech	3T	T1w	4	20	20
D <sub>5</sub>	IXI	3T	T2w	4	20	20

new. The number of training and test images for each domain indicated in the table are explained later while describing the experiments.

**Training Details:** While the domain-specific BN layers can be incorporated in any standard CNN, we work with the widely used U-Net [2] architecture with minor alterations. Namely, our network has a reduced depth with three max-pooling layers and a reduced number of kernels: 32, 64, 128, 256 in the convolutional blocks on the contracting path and 128, 64, 32 on the upscaling path. Also, bilinear interpolation is preferred to deconvolutional layers for upscaling in view of potential checkerboard artifacts [23]. The network is trained to minimize the dice loss, as introduced in [24] to reduce sensitivity to imbalanced classes. Per image volume, the intensities are normalized by dividing by their 98<sup>th</sup> percentile. The initial network trains in about 6 h, while the domain-specific BN modules can be updated for a new domain in about 1 h, on a Nvidia Titan Xp GPU.

**Experiments:** We train three types of networks, as described below.

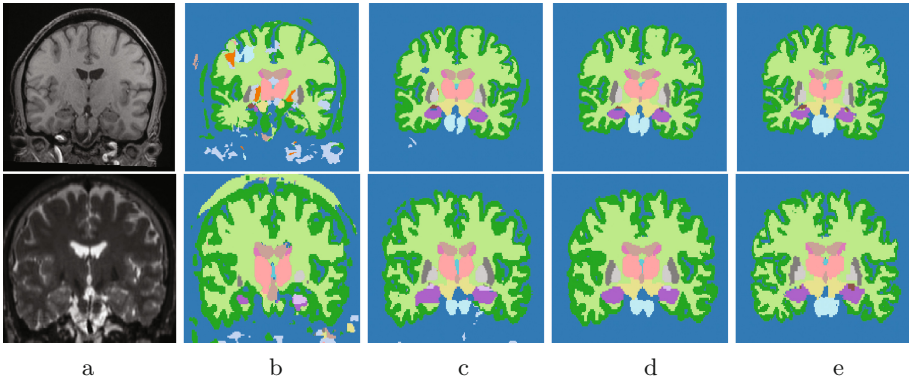
- Individual networks  $N_d$ : Trained for each domain  $d$ , with  $n_{train}^{scratch}$  training images (see Table 1). For the known domains ( $D_1$ ,  $D_2$ ,  $D_3$ ), the accuracy of  $N_d$  serves as a baseline that the other networks with shared parameters must preserve. For the new domains ( $D_4$ ,  $D_5$ ), the performance of  $N_d$  is the benchmark that we seek to achieve by training on much fewer training examples ( $n_{train}$ ) and using the knowledge of the previously learned domains.
- A shared network  $N_{123}$ : Trained on  $D_1$ ,  $D_2$ ,  $D_3$  with  $n_{train}$  images, with all parameters shared including the BN layers,  $bn_s$ . In contrast to the training regime of  $N_{1,2,\dots,d}^{bn}$  described in Sect. 2, while training  $N_{123}$  each batch randomly contains images from all domains to ensure that the shared BN parameters can be tuned for all domains. Histogram equalization [25] is applied to a new domain  $D_d$  before being tested  $N_{123}$ . For adapting  $N_{123}$  to  $D_d$ , all parameters are fine-tuned with  $n_{train}$  images of the new domain and the modified network is referred to as  $N_{123 \rightarrow d}$ .
- A lifelong multi-domain learning network  $N_{123}^{bn}$ : Trained on  $D_1$ ,  $D_2$ ,  $D_3$ , with shared convolutional layers and domain-specific BN layers. The updated network after extending  $N_{123}^{bn}$  for a new domain  $D_d$  according to the procedure described in Sec. 2 is called  $N_{123,k^* \rightarrow d}$ , where  $k^*$  is the closest domain to  $D_d$ .

**Table 2.** Segmentation dice scores for different domains for the three different types of networks, trained as explained in the experiments section.

Network	Test	BN	Thal	Hipp	Amyg	Ventr	Caud	Puta	Pall	Avg
N <sub>1</sub>	D <sub>1</sub>	bn <sub>1</sub>	0.919	0.861	0.849	0.901	0.9	0.887	0.747	0.866
N <sub>2</sub>	D <sub>2</sub>	bn <sub>2</sub>	0.912	0.84	0.836	0.891	0.889	0.876	0.736	0.854
N <sub>3</sub>	D <sub>3</sub>	bn <sub>3</sub>	0.913	0.872	0.81	0.944	0.864	0.879	0.853	0.876
N <sub>4</sub>	D <sub>4</sub>	bn <sub>4</sub>	0.924	0.879	0.853	0.933	0.912	0.9	0.851	0.893
N <sub>5</sub>	D <sub>5</sub>	bn <sub>5</sub>	0.884	0.79	0.773	0.803	0.793	0.818	0.791	0.81
N <sub>123</sub>	D <sub>1</sub>	bn <sub>s</sub>	0.909	0.846	0.824	0.891	0.878	0.877	0.745	0.853
N <sub>123</sub>	D <sub>2</sub>	bn <sub>s</sub>	0.888	0.838	0.815	0.876	0.863	0.86	0.701	0.834
N <sub>123</sub>	D <sub>3</sub>	bn <sub>s</sub>	0.905	0.851	0.792	0.938	0.863	0.873	0.828	0.864
N <sub>123</sub>	D <sub>4</sub>	bn <sub>s</sub>	0.745	0.249	0.057	0.787	0.428	0.324	0.071	0.38
N <sub>123</sub>	D <sub>4,HistEq</sub>	bn <sub>s</sub>	0.641	0.428	0.175	0.754	0.628	0.579	0.303	0.501
N <sub>123→4</sub>	D <sub>4</sub>	bn <sub>s</sub>	0.91	0.856	0.74	0.922	0.894	0.859	0.786	0.852
N <sub>123→4</sub>	D <sub>1</sub>	bn <sub>s</sub>	0.869	0.809	0.773	0.867	0.861	0.722	0.667	0.795
N <sub>123→4</sub>	D <sub>2</sub>	bn <sub>s</sub>	0.676	0.418	0.512	0.105	0.635	0.4	0.411	0.451
N <sub>123→4</sub>	D <sub>3</sub>	bn <sub>s</sub>	0.801	0.762	0.65	0.753	0.728	0.715	0.772	0.74
N <sub>123</sub>	D <sub>5</sub>	bn <sub>s</sub>	0.418	0.178	0.182	0.438	0.268	0.197	0.025	0.244
N <sub>123</sub>	D <sub>5,HistEq</sub>	bn <sub>s</sub>	0.294	0.143	0.16	0.437	0.261	0.293	0.01	0.228
N <sub>123→5</sub>	D <sub>5</sub>	bn <sub>s</sub>	0.861	0.777	0.761	0.799	0.76	0.796	0.741	0.785
N <sub>123→5</sub>	D <sub>1</sub>	bn <sub>s</sub>	0.267	0.022	0.173	0.004	0.05	0.002	0.004	0.075
N <sub>123→5</sub>	D <sub>2</sub>	bn <sub>s</sub>	0.574	0.574	0.564	0.739	0.657	0.521	0.526	0.594
N <sub>123→5</sub>	D <sub>3</sub>	bn <sub>s</sub>	0.147	0.029	0.16	0.006	0.114	0.039	0.003	0.071
N <sub>123</sub> <sup>bn</sup>	D <sub>1</sub>	bn <sub>1</sub>	0.916	0.852	0.84	0.894	0.893	0.884	0.729	0.858
N <sub>123</sub> <sup>bn</sup>	D <sub>2</sub>	bn <sub>2</sub>	0.91	0.853	0.843	0.887	0.882	0.873	0.749	0.857
N <sub>123</sub> <sup>bn</sup>	D <sub>3</sub>	bn <sub>3</sub>	0.911	0.868	0.818	0.944	0.867	0.879	0.846	0.876
N <sub>123</sub> <sup>bn</sup>	D <sub>4</sub>	bn <sub>1</sub>	0.621	0.288	0.218	0.173	0.676	0.576	0.457	0.43
N <sub>123</sub> <sup>bn</sup>	D <sub>4</sub>	bn <sub>2</sub>	0.162	0	0.001	0.001	0.04	0.017	0	0.032
N <sub>123</sub> <sup>bn</sup>	D <sub>4</sub>	bn <sub>3</sub>	0.721	0.271	0.305	0.549	0.569	0.515	0.297	0.461
N <sub>123,3→4</sub> <sup>bn</sup>	D <sub>4</sub>	bn <sub>4</sub>	0.878	0.83	0.772	0.907	0.875	0.852	0.772	0.841
N <sub>123</sub> <sup>bn</sup>	D <sub>5</sub>	bn <sub>1</sub>	0.001	0.019	0.062	0.008	0.004	0	0	0.013
N <sub>123</sub> <sup>bn</sup>	D <sub>5</sub>	bn <sub>2</sub>	0.354	0.123	0.268	0.225	0.407	0.276	0.366	0.288
N <sub>123</sub> <sup>bn</sup>	D <sub>5</sub>	bn <sub>3</sub>	0	0.003	0.031	0.001	0	0	0	0.005
N <sub>123,2→5</sub> <sup>bn</sup>	D <sub>5</sub>	bn <sub>5</sub>	0.774	0.687	0.687	0.761	0.669	0.714	0.713	0.715

**Results:** All networks are evaluated based on their mean Dice score for  $n_{test}$  images from the appropriate domain (see Table 1). Quantitative results of our experiments are shown in Table 2. The findings can be summarized as follows:

- $N_{123}$  preserves the performance of  $N_1, N_2, N_3$ . Thus, a single network can learn to segment multiple domains, provided sufficient training data is available from all the domains at once. However, its performance severely degrades for unseen domains  $D_4$  and  $D_5$ . Histogram equalization (denoted by  $D_{d,HistEq}$ ) to the closest domain is unable to improve performance significantly, while fine-tuning the network for the new domains causes catastrophic forgetting [20], that is, degradation in performance on the old domains.
- $N_{123}^{bn}$  also preserves the performance of  $N_1, N_2, N_3$ . For a new domain  $D_4$ , using the  $bn_3$  parameters of the trained  $N_{123}^{bn}$  leads to the best performance. Thus, we infer that  $D_3$  is the closest to  $D_4$  among  $D_1, D_2, D_3$ . After fine-tuning the parameters of  $BN_3$  to obtain those of  $BN_4$ , the dice scores for all the structures improve dramatically and are comparable to the performance of  $N_4$ . Crucially, as the original  $bn_k$  for  $k = 1, 2, 3$  are saved, the performance on  $D_1, D_2, D_3$  in the updated network  $N_{123,3-4}^{bn}$  is exactly the same as in  $N_{123}^{bn}$ . Similar results can be seen for the other new domain,  $D_5$ . The improvement in the segmentations for new domains after fine-tuning the BN parameters can also be observed qualitatively in Fig. 2.



**Fig. 2.** Qualitative results: (a) images from domains  $D_d$ , segmentations predicted by (b)  $N_{123}^{bn}$ ,  $bn_{k^*}$ , (c)  $N_{123,k^* \rightarrow d}^{bn}$ ,  $bn_d$ , (d)  $N_d$  and (e) ground truth annotations, with  $\{d, k^*\}$  as  $\{4, 3\}$  (top) and  $\{5, 2\}$  (bottom).

## 4 Conclusion

In this article, we presented a lifelong multi-domain learning approach to learn a segmentation CNN that can be used for related MR modalities and across scanners/protocols. Further, it can be adapted to new scanners or protocols with only a few labelled images and without degrading performance on the previous scanners. This was achieved by learning batch normalization parameters for each scanner, while sharing the convolutional filters between all scanners. In future

work, we intend to investigate the possibility of extending this approach to MR modalities that were not present during the initial training.

To the best of our knowledge, this is the first work to tackle the lifelong machine learning problem for CNNs in the context of medical image analysis. We believe that this may set an important precedent for more research in this vein to handle data distribution changes which are ubiquitous in clinical data.

## References

1. Fischl, B.: Freesurfer. *Neuroimage* **62**(2), 774–781 (2012)
2. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) *MICCAI 2015, Part III*. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). [https://doi.org/10.1007/978-3-319-24574-4\\_28](https://doi.org/10.1007/978-3-319-24574-4_28)
3. Despotović, I., et al.: MRI segmentation of the human brain: challenges, methods, and applications. *Comput. Math. Methods Med.* **2015**, 23 (2015)
4. Preboske, G., et al.: Common MRI acquisition non-idealities significantly impact the output of the boundary shift integral method of measuring brain atrophy on serial MRI. *Neuroimage* **30**(4), 1196–1202 (2006)
5. Jovicich, J., et al.: Reliability in multi-site structural MRI studies: effects of gradient non-linearity correction on phantom and human data. *Neuroimage* **30**(2), 436–443 (2006)
6. Pan, S., et al.: A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **22**(10), 1345–1359 (2010)
7. Dredze, M., et al.: Multi-domain learning by confidence-weighted parameter combination. *Mach. Learn.* **79**(1–2), 123–149 (2010)
8. Thrun, S.: Lifelong learning algorithms. In: Thrun, S., Pratt, L. (eds.) *Learning to Learn*, pp. 181–209. Springer, Boston (1998). [https://doi.org/10.1007/978-1-4615-5529-2\\_8](https://doi.org/10.1007/978-1-4615-5529-2_8)
9. Zhuge, Y., et al.: Intensity standardization simplifies brain MR image segmentation. *Comput. Vis. Image Underst.* **113**(10), 1095–1103 (2009)
10. Weisenfeld, N., et al.: Normalization of joint image-intensity statistics in MRI using the Kullback-Leibler divergence. In: *IEEE International Symposium on Biomedical Imaging: Nano to Macro*, vol. 2004, pp. 101–104. IEEE (2004)
11. Han, X., et al.: Atlas renormalization for improved brain MR image segmentation across scanner platforms. *IEEE Trans. Med. Imaging* **26**(4), 479–486 (2007)
12. Van Oprobroek, A., Ikram, M.A., Vernooij, M., De Bruijne, M.: Transfer learning improves supervised image segmentation across imaging protocols. *IEEE Trans. Med. Imaging* **34**(5), 1018–1030 (2015)
13. Cheplygina, V., Pena, I.P., Pedersen, J.H., Lynch, D.A., Sørensen, L., de Bruijne, M.: Transfer learning for multi-center classification of chronic obstructive pulmonary disease. *IEEE J. Biomed. Health Inform.* **22**(5), 1486–1496 (2018)
14. Kamnitsas, K., et al.: Unsupervised domain adaptation in brain lesion segmentation with adversarial networks. In: Niethammer, M., et al. (eds.) *IPMI 2017*. LNCS, vol. 10265, pp. 597–609. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-59050-9\\_47](https://doi.org/10.1007/978-3-319-59050-9_47)
15. Bilen, H., Vedaldi, A.: Universal representations: the missing link between faces, text, planktons, and cat breeds. arXiv preprint [arXiv:1701.07275](https://arxiv.org/abs/1701.07275) (2017)



16. Ioffe, S., et al.: Batch normalization: accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning, pp. 448–456 (2015)
17. Li, Y., Wang, N., Shi, J., Liu, J., Hou, X.: Revisiting batch normalization for practical domain adaptation. arXiv preprint [arXiv:1603.04779](https://arxiv.org/abs/1603.04779) (2016)
18. Carlucci, F., Porzi, L., Caputo, B., Ricci, E., Bulo, S.: Autodial: Automatic domain alignment layers. In: IEEE International Conference on Computer Vision (ICCV 2017), pp. 5077–5085. IEEE (2017)
19. Rebuffi, S., et al.: Learning multiple visual domains with residual adapters. In: Advances in Neural Information Processing Systems, pp. 506–516 (2017)
20. French, R.: Catastrophic forgetting in connectionist networks. *Trends Cogn. Sci.* **3**(4), 128–135 (1999)
21. Van Essen, D., et al.: The WU-Minn human connectome project: an overview. *Neuroimage* **80**, 62–79 (2013)
22. Di Martino, A., et al.: The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism. *Mol. Psychiatry* **19**(6), 659 (2014)
23. Odena, A., et al.: Deconvolution and checkerboard artifacts. *Distill* (2016)
24. Milletari, F., et al.: V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: 2016 Fourth International Conference on 3D Vision (3DV), pp. 565–571. IEEE (2016)
25. Nyúl, L., et al.: New variants of a method of MRI scale standardization. *IEEE Trans. Med. Imaging* **19**(2), 143–150 (2000)