Robustness to Domain Shifts in MRI for Deep Learning-based Methods: A Review

Ryan Pollitt[1]

***Abstract*—Deep learning-based approaches have seen a lot of success in the space of Magnetic Resonance Imaging from segmentation to real-time registration. However, these methods often fail to generalize to different domains, that is to say scans from different scanners, sequences or populations. We review two classes of approaches to counteract these so-called domain shifts in deep learning for MRI data: data harmonization and domain generalization, and discuss their pros and cons. Data harmonization removes domain-specific information from MR images themselves, whereas domain generalization trains a neural network to be robust to a wide variety of input images from different domains. Five papers were found for data harmonization and sixteen papers for domain generalization. Based on these papers, we conclude that both data harmonization and domain generalization are viable for small expected domain shifts. In practice, the extent of domain shifts will often be unknown prior to deployment of the neural network or will be too large. In this more realistic case, we determine that domain generalization offers better generalization capabilities than data harmonization. On the other hand, data harmonization can be used to remove domain-specific information from new data, making it possible to use already trained task networks (e.g. a segmentation network) without having to retrain on the new domain or requiring labelled data from this domain. Both methods therefore have useful applications in different scenarios.**

***Index Terms*—Data harmonization, deep learning, domain generalization, domain shift, MRI**

I. INTRODUCTION

MEDICAL imaging modalities like Magnetic Resonance Imaging (MRI), Computed Tomography (CT), Positron Emission Tomography, X-rays and ultrasound allow radiologists to analyse the structure and/or function of e.g. internal organs and the musculoskeletal system. A lot of data is generated using these modalities, especially with MRI, which allows for the acquisition of 3D volumes with different contrasts capable of highlighting certain pathologies or abnormalities. These data are subsequently viewed and analysed by radiologists in clinical practice or analysed for research purposes.

The processing and analysis of large amounts of data in the clinic or for research is labour- and time-intensive. A promising approach for time-efficient and powerful processing and analysis of this data is deep learning (DL). DL concerns the training of neural networks, often convolutional neural networks (CNNs) in the medical imaging domain, which are capable of learning useful features from input imaging data and subsequently using these to produce a target output. Training a network to produce a target output from a given input is called supervised learning, which is what most DL approaches use [1]. An example of this could be a network that is trained to reproduce segmentation maps of tumours, which could subsequently be used for automatic tumour detection and/or volumetric analysis. Other examples of DL applications beyond segmentation include:

- Image synthesis; e.g. to balance a dataset to include more healthy/pathological cases or to create additional useful images, like generating synthetic CT images from MR images
- Image registration; allowing for real-time affine and/or deformable registration of images for e.g. motion correction
- Image reconstruction; allowing for MR image reconstruction from fewer k-space samples
- Image super resolution; creating higher resolution images
- Classification; classifying images into e.g. healthy vs. pathological cases [1].

An assumption is often made that the unseen/test data that the trained network is applied to is similar to the training data. In practice, however, this assumption limits the application of DL-based methods, because they only perform well on test data similar to the training data. Once a so-called domain shift occurs in the test domain, meaning that the statistics of the test data are different from the ones of the training data, DL-based methods often fail to generalize, which is detrimental to the wide-scale application of DL-based technologies [2]. Domain shifts within MRI can result from different factors, e.g.:

- Differences in sequence parameters like repetition time, echo time, flip angle, resolution, and the type of sequence used (e.g. spin echo or gradient echo)
- Differences between scanner hardware and software, e.g. 1.5 T field strength vs. 3 T or Siemens vs. Philips [3]
- Differences in subject/patient populations' distributions of sex, age, and pathology (or absence thereof) [2].

In this review a domain is defined as (data from) a

[1] Student of the Master's programme Medical Imaging at Utrecht University

combination of a specific sequence type, scanner type and population. Two domains are considered different if one or more of these three variables are different, resulting in a domain shift. Multiple methods exist to alleviate the performance drop resulting from a domain shift. One such method is transfer learning, in which a network is trained on the training data and subsequently finetuned/retrained on a smaller dataset from the test domain. Transfer learning is often done by freezing part of the network's parameters and only training a subset of them on a small dataset of labelled data from the test domain [4]. Labelled data in this context could refer to e.g. segmentation maps paired with the corresponding MR images, or CT scans corresponding to MR images to train a network for synthetic CT generation. Although transfer learning only requires a small amount of labelled data from the test domain, it is still suboptimal in practice, because it requires each site that wants to deploy the neural network to retrain the network and to acquire additional labelled data.

Other methods also exist, which avoid adapting the neural network to each test domain: data harmonization (DH) and domain generalization (DG), which are discussed in the following paragraphs. DH seeks to reduce sources of variability between different domains. More specifically, most DH methods in the space of MR imaging focus on a post-processing step in which the MR images are altered to remove domain-dependent features. Older methods were based on histograms/global image statistics, but these do not address local domain-specific variations in the images [5]. More advanced techniques that can take local information into account are DL-based methods, which often learn an image-to-image mapping from multiple domains to a single reference domain. This mapping can be learned by a CNN in a supervised manner, where a group of subjects is scanned with different scanners and/or with different sequences. By removing domain-specific information using the trained harmonization CNN to map from multiple domains to a reference domain, a second CNN (e.g. for segmentation) trained on the reference domain can theoretically be applied to an arbitrary number of domains, provided that the harmonization network has been trained to translate these to the reference domain. DH methods also exist that do not require a dataset with overlap. This is called unsupervised DH and will be the only type of DH considered in this study, as acquiring images with an overlap cohort is often impractical [6].

DH could be beneficial for removing site/scanner-specific features in multi-centre studies, which allow for a much larger dataset to be constructed with more statistical power [7]. Removing site/scanner-specific features is necessary, because the site at which a subject is scanned can greatly influence metrics derived from the images [8].

Finally, robustness to domain shifts can also be achieved using DG, which differs from DH, because it does not require a harmonization network to bring images to a reference domain, but instead tries to generalize the task network itself to multiple domains.

DG can be achieved in two ways: 1) by training a network in such a way that it learns domain-independent features and 2) by increasing the diversity of the data (either by augmentations or data synthesis) during training. The main differences between these two are that the second method does not change anything about the training itself and only implicitly forces the network to learn domain-independent features, whereas the first method explicitly forces the network to learn domain-independent features by using advanced training techniques. The latter can be achieved with domain adversarial learning or meta-learning, which are two advanced training methods aimed at forcing the network to learn domain-independent features. A high level overview of DH and the different DG methods is given in Fig. 1 along with the sections in which they are discussed, where different domains are indicated with different colours.

This review will discuss **data harmonization and domain generalization** applied to structural MRI to investigate the potential of these methods to counteract the domain shift problem. We focus on structural MRI, in which images reflect differences in tissue parameters (T_1, T_2, ρ), as these images are acquired most often in the clinic. The goal is to give an overview of the different methods that have been applied in this context and to discuss pros and cons of each method.

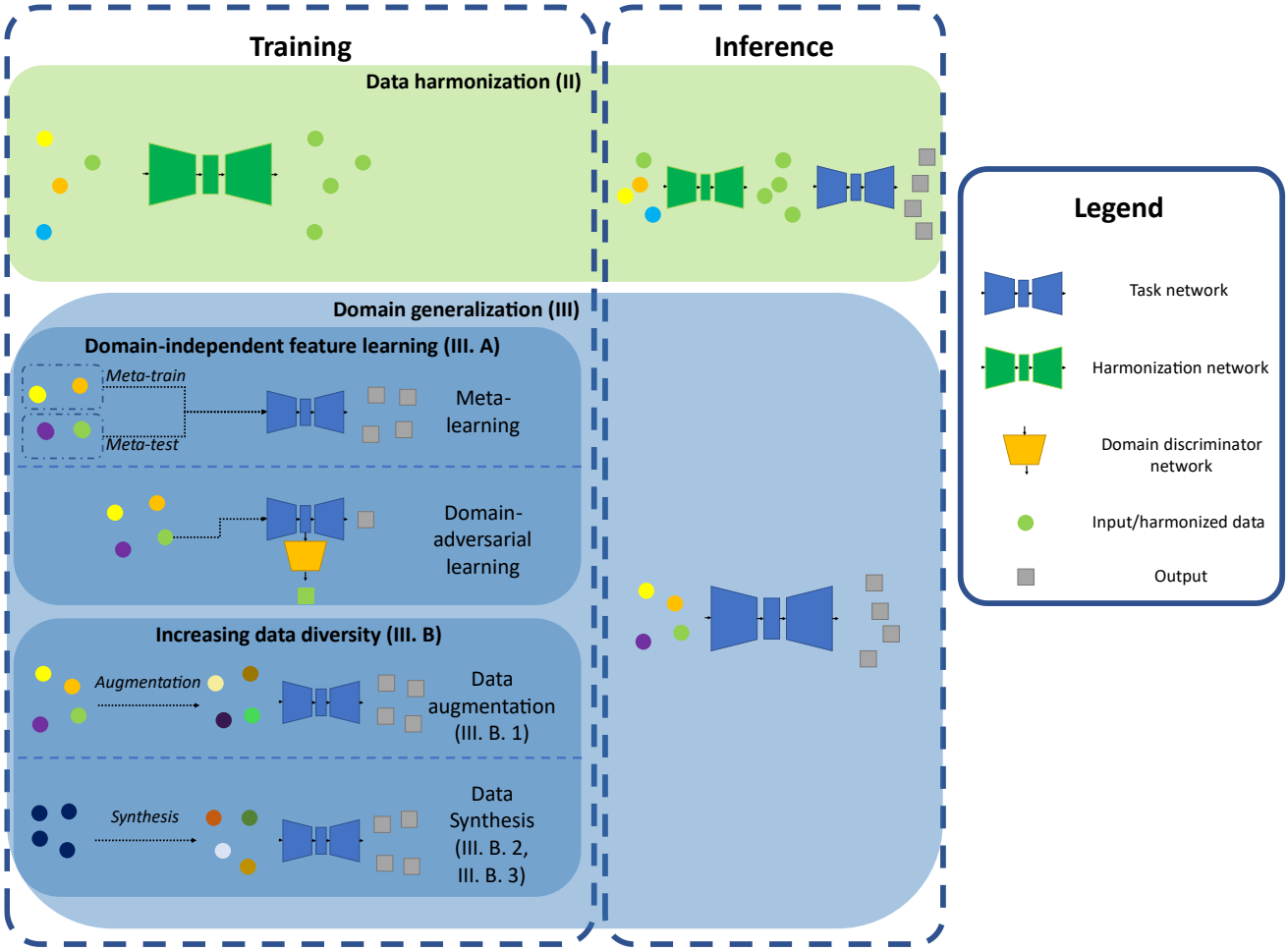

Fig. 1. High level overview of DH and the different DG methods during training of the network and when used for inference. The different colours of the circles represent different domains.

I. A. Selection criteria

To find relevant literature Google Scholar was used with the search terms "MRI harmonization", "MRI domain generalization" and "MRI domain shift". The exclusion criteria were:

- Papers that required labelled data for the unseen domain (as in transfer learning) or rescans of the same subjects at different sites, as this type of data is often absent or impractical to acquire
- Papers that applied to diffusion weighted imaging (DWI), because DWI data is quite different from the more often clinically acquired structural MRI and therefore lends itself to harmonization techniques not applicable to structural MRI
- Papers with zero citations, unless they were published in the past six months
- Papers that did not use MRI data
- Papers that did not employ DH or DG

If papers mentioned other DG or DH methods, these were also checked against the exclusion criteria and included if relevant. Google Scholar gave on the order of 10,000-100,000 results per search term. The search was stopped if 40 results in a row did not employ DH (for the search terms "MRI harmonization - diffusion" and "MRI domain shift") or DG (for the search terms "MRI domain generalization" and "MRI domain shift") under the assumption that more relevant results come first in the search. Using these selection criteria five papers focusing on DH and sixteen papers that studied DG were found. An overview of all 21 methods is given in Table A1, which the reader can refer to for a general overview or to see which network architecture was used in each paper, which is often omitted in the main text for brevity.

The following sections are structured as follows: section II reviews relevant literature about DH and section III discusses DG. Both methods and their subcategories are discussed and compared in section IV, with a final conclusion in section V.

II. Data harmonization

The following subsection describes two ways of training CNNs for DH: using Generative Adversarial Networks (GANs) and encoder-decoder architectures.

II. A. GAN-based harmonization

This subsection describes two examples of methods that performed DH using GANs, though more examples exist. GANs employ two types of networks, a generator and a discriminator [9]. Given a random noise vector or input image, the generator tries to generate a realistic image to fool the discriminator, which tries to discriminate between real images and fake images generated by the generator. The generator is optimized to maximize the probability that the discriminator assigns the generated image as being real, while the discriminator is optimized to minimize this same probability and to maximize the probability of predicting a real image as being real.

This idea is expanded upon in CycleGAN [10], which employs two generators and two discriminators to cycle between two types of images (e.g. T1-weighted to T2-weighted images) and enforces a cycle consistency, where mapping from e.g. a T1-weighted (T1w) image to a T2-weighted (T2w) image using the first generator and mapping back to the T1w image using the second generator should result in the same image being reconstructed. Due to its cyclic nature, CycleGAN does not need paired data.

The authors of the papers in this subsection trained a network for the segmentation of brain structures [11] and age estimation based on brain scans [2], respectively. In brain age estimation, the age of a subject is estimated based on a scan of their brain. This age estimation functions as a biomarker of brain pathology if significantly different from biological age [2]. The authors of these two papers subsequently trained a GAN variant, CycleGAN in [11] and StarGAN v2 in [2], to harmonize scans acquired with different scanners and similar sequences (and from a different population in [11]) to the training domain so they could apply their segmentation/age estimation networks. The next paragraphs discuss these two papers in more detail.

In [11] the authors trained a CNN to segment (subregions of) the amygdala from 14 healthy subjects aged 8.5–43.4 years (mean of 28.9 years) scanned with a 3D inversion-recovery prepared fast gradient-echo T1-weighted sequence. After training their segmentation CNN, they tested its generalizability to scans of children/adolescents aged 9-18 years scanned with relatively similar T1w protocols, but scanned at 13 sites with different 3 T MRI scanners. Additionally, these subjects suffered a traumatic brain injury (TBI) 1-2 years before scanning, giving very heterogeneous data.

To bridge this domain shift resulting from differences in age range, type of scanner, sequence used and pathology they used a CycleGAN to learn a transformation to the domain of the data that the segmentation CNN was trained on. A limitation of the study is that it is unclear if they used separate generators/discriminators for each of the 13 sites, but it seems they consider all of the TBI scans as one domain and the training domain as the reference domain.

After applying the generator that transforms the images to the reference domain they directly applied the segmentation network trained on the reference domain data and achieved a Dice Similarity Coefficient (DSC) of 0.755±0.067 for amygdala segmentation compared to 0.428±0.218 on the unharmonized data and 0.760±0.096 when the segmentation network was trained exclusively on the TBI scans with 7-fold cross-validation.

Therefore, they showed that in this case transforming data to a different reference domain on which a CNN was trained versus training on the data itself can lead to similar results. Note that the similarity in performance is related to the TBI data being much more heterogeneous than the data of the healthy subjects. This seems to make it harder for the network to learn from the TBI data, which is indicated by the fact that training and evaluating on the reference dataset of healthy subject with cross-validation gave a much higher DSC of 0.906±0.019.

The authors of the second paper trained a CNN for brain image-based age estimation [2]. The authors trained this network on a reference domain of T1w Magnetization Prepared Rapid Gradient Echo (MPRAGE) images scanned using a 1.5 T Siemens scanner. To apply this network to five other domains consisting of T1w MPRAGE and Spoiled Gradient Echo (SPGR) sequences scanned using different scanners from Siemens, GE and Philips at either 1.5 T or 3 T field strength, they used a GAN called StarGAN v2 (henceforth referred to as StarGAN) [12].

In contrast to CycleGAN, StarGAN does not need to train $N(N-1)$ generators to translate between N different domains, but uses a shared generator to translate between all N domains. Another difference compared to CycleGAN is that StarGAN explicitly separates image content from the image domain. Conceptually, this means that e.g. two T1w images of the same patient and same anatomy acquired with different scanners should result in the same image content, but two distinct domain encodings. The domain encoder is a separate trainable network along with the discriminator and generator. By extracting the domain encoding of an image from the reference domain and supplying this to the generator along with the image content of an image from a different domain, harmonization to the reference domain can be achieved.

The generator and domain encoder were used at inference time to harmonize the five other domains to the reference domain. Harmonization was achieved by inputting the images from these five domains to the generator and applying the reference domain code, which was extracted by the encoder from images from the reference domain. An example of harmonizing from one of the five domains to the reference domain is shown in Fig. 2. The figure shows a few slices of two different subjects at similar locations in the brain and the corresponding Mean Absolute Error (MAE) for age estimation of this dataset before and after harmonization.

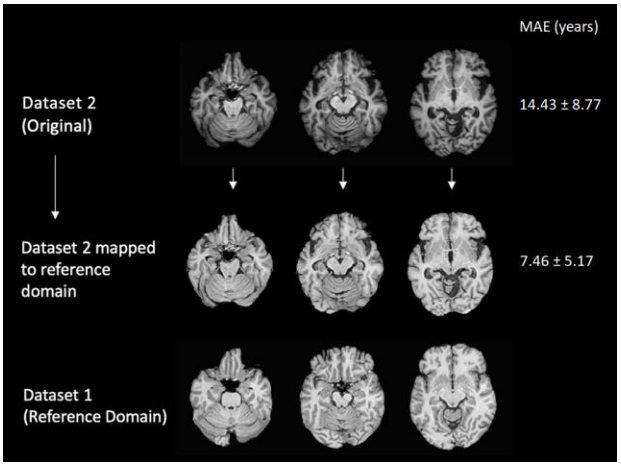

Fig. 2. Examples of harmonizing from one domain (Dataset 2) to a reference domain (Dataset 1) with the corresponding Mean Absolute Error (mean ± standard deviation) for brain age estimation of Dataset 2 without and with harmonization. Adapted from [2].

The harmonization of all five domains resulted in a Mean Absolute Error (MAE) of 7.21 years and Pearson correlation coefficient (PCC) of 0.779 between predicted- and real age compared to MAE's and PCC's of 11.86 years/0.341 and 15.81 years/0.299 for the histogram matched- and unharmonized images, respectively. The authors therefore demonstrated a large performance increase compared to traditional histogram matching when using DL-based harmonization methods and showed the even larger benefit of using this technique compared to using no harmonization.

II. B. Encoder-decoder-based harmonization

All three methods in this subsection use basic components of autoencoders (AEs), in which an encoder is tasked with creating a (lower dimensional) representation of the input, which is subsequently reconstructed by a decoder. Both the encoder and decoder are often CNNs. Table I gives a short overview of which anatomy was used in each study, across what domains the authors harmonized images and if a downstream task was applied to the images for extra validation of the harmonization performance. As opposed to the previous two papers the three papers of this subsection focused mostly on the harmonization itself, with only one using a non-DL downstream task for validation.

TABLE I
OVERVIEW OF METHODS IN THIS SUBSECTION, WHAT ANATOMY THEY STUDIED, WHICH DOMAINS THEY HARMONIZED ACROSS AND IF A DOWNSTREAM TASK WAS EVALUATED FOR VALIDATION

Ref.	Anatomy	Domains	Downstream task
[13]	Brain	Population, scanner and sequence	No
[14]	Brain	Scanner and sequence	No
[15]	Brain	Scanner	Yes

In [13] the authors trained a network to translate between T1w and T2w brain images from three different scanners and sites, with healthy- and multiple sclerosis subjects. Similarly to [2] they explicitly separated image content (anatomical features) from domain (sequence and scanner type). This was achieved by forcing the encoder network to output an anatomical feature map and a contrast component. The anatomical feature maps had the same height and width as the input image, but with five channels approximating a one-hot encoded feature map. The one-hot encoding means that one of the channels has a value of approximately 1, while the other four have a value of approximately 0, encoding the presence (1) or absence (0) of the feature encoded by each channel. A Straight-Through Gumbel-Softmax layer enforced the approximation of a one-hot encoded feature map such that the network could learn five relevant features. By constraining these feature maps to be one-hot encoded with only five channels the authors tried to prevent the network from encoding domain-specific information in the anatomical feature maps. The contrast code on the other hand was a single learnable scalar, determined by the network. For example, T1w scans from Site A were attributed a mean scalar value of -2300, T1w scans from Site B a mean value of -3700 and T2w scans from Site A a mean value of -400 by the network.

The encoder was a U-Net architecture [16], which gave the anatomical feature map and contrast code. The decoder was also a U-Net, which took in the contrast component and anatomical feature maps and was tasked with reconstructing the anatomical information with the requested contrast. More specifically, the networks were tasked with encoding and reconstructing both the T1w and T2w images of the same patient, where the encoded anatomical information from both scans could be combined with the encoded contrast code of both scans for a total of four combinations. These combinations resulted in two images that were compared to the T1w image and two that were compared to the T2w image with a Mean Square Error loss. Additionally, they enforced a cosine similarity on the anatomical feature representations, which should be the same for the T1w- and T2w images.

After training the networks, T1w images of multiple sclerosis subjects that were scanned at two of the sites were harmonized from one site to the other by encoding the anatomical information from scans at site A and taking the average contrast encoding value of scans from site B and supplying these to the decoder. After harmonization of these twelve subjects the mean structural similarity index between the images from the two sites went up from 0.845 to 0.923.

The same research group proposed a follow-up to the aforementioned paper in [14] with three key differences. Firstly, the contrast encoding was performed on a different slice than the anatomical information encoding, such that it was impossible for the network to encode anatomical information in the contrast component. Secondly, they applied a discriminator to the anatomical information encoding, where the discriminator was tasked with learning if the anatomical information came from a single site A or not. They trained the

encoder in an adversarial fashion such that its outputs did not contain information about the input's site. This component was added, because in the previous setup from [13], the network would still theoretically be able to encode which site the data came from. This is because enforcing the cosine similarity between the anatomical features from two scans of the same patient scanned at a single site only discourages the network from encoding domain-specific information (e.g. T1w vs. T2w) within each site and not across sites. Finally, two completely separate encoders were used to encode the anatomical information and contrast as opposed to using one encoder to encode both components.

Harmonization of brain scans was tested between four different domains consisting of scans from either 1.5 T or 3T Siemens scanners scanned using slightly different sequence parameters in each of the four domains. The method significantly outperformed CycleGAN, their older method from [13], and histogram matching in a majority of the metrics, which were Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) for a test set of T1w- and T2w brain scans of subjects scanned at multiple sites.

The aforementioned results were achieved on domains that were used during training of the harmonization network. On the other hand, the authors also showed a method to generalize their harmonization to new domains not seen during training by finetuning small parts of the network. To demonstrate this they trained a new harmonization network with T1w- and T2w brain scans from two sites using the same 3 T Philips scanner, but different sequence parameters. They then applied this trained network to a third domain consisting of T1w brain scans obtained with a different 3 T Philips scanner and different sequence parameters.

To finetune the trained networks to the new domain they froze all network parameters except the last few layers of the anatomical information encoder. These were trained using the frozen anatomical information discriminator to bring the anatomical information distribution closer to that seen during training to remove site-specific information. The authors showed that this finetuning improved PSNR and SSIM compared to applying the network without finetuning, while only requiring unlabelled data from the test domain.

The final paper discussed in this subsection, which also used an encoder-decoder-based architecture and a discriminator, is [15]. To evaluate their approach they used healthy brain scans from six different domains using scanners from GE, Philips and Siemens operating at either 1.5 T or 3 T field strength scanned using either a 3D T1w MPRAGE (Philips and Siemens) or T1w SPGR (GE). They derived radiomic features from all scans and because all scans were of healthy subjects they expected a reduction in statistical differences between domains after harmonization.

The authors argued that training a network using paired T1w- and T2w data from the same scanner like in [13] and [14] and subsequently using this trained network to harmonize between scanners is not ideal as it requires paired data and assumes that intra- (e.g. T1w vs. T2w) and inter-scanner (e.g. 1.5 T vs. 3 T) differences are interchangeable. By dropping this assumption and the need for paired data, they could not use the separate anatomical feature maps and contrast components as in [13] and [14]. This leads to the following two key differences between this method and the previous two methods: 1) it mostly concerns domain differences resulting from different types of scanners with sequences more closely aligned across scanners (all T1w instead of T1w and T2w) and 2) it does not explicitly separate anatomical information from contrast.

Instead, the images were encoded into a latent space without a separation between anatomical information and contrast, on which a discriminator acted to learn which domain the latent space represents, while the encoder tried to maximally confuse the discriminator. The purpose of this setup was that the encoder learned to encode domain-agnostic information only.

The authors of [15] also criticized the use of image-level discriminators, which were used in e.g. [2] and [11], because these risk changes in content and feature hallucination [17]. To show the benefit of foregoing an image-level discriminator they quantified the change in/loss of content by translating from one of the six domains to another using the harmonization network and then plugging the output back into the network to harmonize back to the original domain. The expectation was that if content was lost/changed, this would not be recoverable in the translation back to the original domain.

To quantify this, they calculated the SSIM between the original image of each domain and the same image auto-encoded (translated to its own domain) and the SSIM between the original image and the same image after cycling through a different domain and back to the original domain. The difference between these two SSIMs should then be 0 if no content is lost. The authors showed that this difference was 0.0021 without an image-level discriminator and three times higher at 0.0062 with an image-level discriminator, confirming that content is lost to a greater degree when an image-level discriminator is used.

To validate their harmonization method they measured statistical differences in five types of textural radiomic features (e.g. grey-level co-occurrence matrix and grey-tone difference matrix) derived from the images of different domains and found that these differed much less statistically after harmonization with their approach compared to a histogram-based normalization approach and no harmonization. They also found no significant differences in these difference reductions between the five domains that were part of training and the held-out test domain, achieving zero-shot harmonization (without retraining). However, one caveat was that the test domain was quite similar to the training domains.

III. Domain generalization

Most DG approaches fall into two categories: they either directly enforce a network to learn domain-independent features using e.g. an adversarial loss component, or they increase data diversity using data synthesis/data augmentations, which indirectly forces a network to learn domain-independent features. The following two subsections will discuss both of these approaches and their applications in MRI.

III. A. Domain-independent feature learning

Domain-independent feature learning explicitly forces neural networks to learn domain-independent features using advanced training strategies, such that a trained network could in theory be applied to domains not seen during training. Five out of the six methods in this subsection use an adversarial loss to achieve this, whereas the sixth method employs meta-learning, which will be discussed last.

The adversarial loss implementations in this subsection are similar to those of the two previously discussed DH methods [14], [15]. The main difference is that instead of being built into a harmonization network, the domain-independent features are enforced in the task network itself. Fig. 3 shows a diagram of the general structure of these approaches. The task network—often consisting of an encoder and decoder—predicts the output $\hat{y}$ from the input x, which is subsequently compared to the ground-truth y via a task-specific loss (e.g. segmentation or classification). Additionally, a domain discriminator is trained to recognize which domain the input came from based on a combination of feature maps from the task network, often the lowest resolution feature maps and/or the final feature maps before the output layer. The discriminator is optimized to minimize the adversarial loss, which could for example be the cross-entropy between the predicted domain and actual domain. The encoder-decoder network on the other hand is optimized to maximize this loss, giving rise to the domain-adversarial training.

Using this training style the idea is that the domain discriminator becomes better and better at recognizing what domain the input data came from, whereas the encoder-decoder network becomes better at fooling the domain discriminator. If the encoder-decoder network manages to maximally confuse the domain discriminator, the feature maps that are fed into the domain discriminator should theoretically be domain-independent. If the final feature maps are domain-independent, the output is also domain-independent. If only the lowest resolution feature maps are domain-independent, the output is only fully domain-independent if the network does not employ skip connections that concatenate feature maps from the encoder to those of the decoder.

This basic idea was used in [18] for disorder identification and disease progression prediction for brain images, in [19] for segmentation of brain images, in [20] for knee image segmentation and in [21], [22] for cardiac image segmentation.

The authors of the first two papers [18], [19] focused on the scenario in which data from both the training and testing domain are available, with labels for the training domain only. In this setting both $\mathcal{L}_{\text{adversarial}}$ and $\mathcal{L}_{\text{task}}$ are computed and optimized with data from the training domain, whereas only $\mathcal{L}_{\text{adversarial}}$ is optimized for data from the testing domain. The other papers address the more realistic scenario in which the test domain is unseen during training.

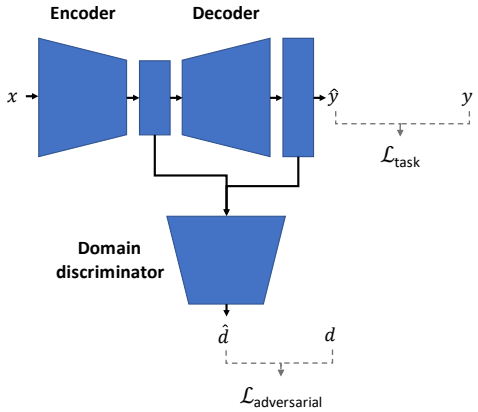

Fig. 3. Overview of the adversarial loss and domain discriminator often used for domain generalization. An input x is forward propagated through the network and a prediction $\hat{y}$ is compared to a ground-truth y via the task loss, while the predicted domain $\hat{d}$ is compared to the real domain d via the adversarial loss.

The authors of the third paper [20] trained a network on two different domains with sagittal 3D double echo steady state MR images of the knee from two different scanners and with slightly differing sequence parameters, of which only one domain was labelled. They trained the network in the same way as outlined in the previous paragraph on these two domains for the segmentation of femoral and tibial cartilage tissues. The trained network was subsequently applied to a third unseen domain, consisting of the same scanner and protocol as the unlabelled domain, but with different patient populations.

Interestingly, the authors of [20] implemented a second generalization technique called mixup, proposed in [23]. Mixup functions as a type of data augmentation and takes random linear interpolations between two inputs and the two corresponding target outputs and feeds these to the network. A classic example would be an image classifier which is fed a linear interpolation between an image of a dog (weighting of e.g. 0.3) and a cat (weighting of e.g. 0.7) and the loss is calculated with a target output of probabilities 0.3 for the dog class and 0.7 for the cat class. By training in this way, the network is forced to behave linearly between samples. Using these linear interpolations a greater input space is explored. Linear behaviour (i.e. a linear interpolation in the input leads to a linear interpolation of the outputs) between samples was thought to be beneficial for generalization from the perspective of Occam's razor, because it is one of the simplest types of behaviours. It was also shown to lead to much smoother decision boundaries in the input space, instead of the sharp decision boundaries from training without mixup [20].

The authors showed that mixup and domain-adversarial learning perform similarly, with domain-adversarial learning sacrificing some performance on the training domain (with labels) for better generalization to the test domain and vice versa for mixup. They argued that mixup is preferable over domain-adversarial learning because it requires less hyperparameter tuning and has a lower computational load [20]. This is because mixup only requires interpolation between

samples, a slight adjustment to the loss calculation, and the choice of a single hyperparameter, which determines how the random interpolations are sampled. For domain-adversarial learning on the other hand, a second network needs to be implemented, requiring a heavier computational load than interpolating data. Domain adversarial learning also requires choosing an optimal domain discriminator network architecture, a potentially different learning rate for the domain discriminator, a choice of the weighting of the adversarial loss component etc.

The last two papers that used domain-adversarial learning are [21], [22], which were both part of a challenge for segmentation of heart structures in data from different scanner types, sites and disease types [24]. Each participant got data from three different sites, of which two were labelled with the final site's data only being available to the challenge organizers. Sequences with relatively similar contrasts were used across sites. The authors of [21], [22] placed #6 and #10 out of 14 submissions. No definite conclusion can be drawn about domain-adversarial learning versus other methods from this, however, because too many other contributing factors like different network architectures, loss functions, the use of ensembles versus single networks, etc. also played into the performance differences. These large differences in methods resulted from the fact that the challenge did not focus strictly on DG and gave participants too much freedom in their approach to be able to draw strong conclusions about the DG methods in isolation. The submission that placed first in this challenge is [25], which will be discussed in the next subsection on data augmentation.

Lastly, a different approach to learn domain-independent features is meta-learning, as proposed in [26], which was used in [27] to train a network to segment the prostate from images from five domains consisting of different sequences and scanner types (field strength and vendor), such that the network generalized to a sixth unseen domain.

In meta-learning the training dataset consists of multiple domains, which are randomly split up into a meta-training set and meta-test set. The meta-training set is used to compute the task-specific loss and the network parameters are temporarily updated according to the loss. Then, the meta-test set is used to compute the task-specific loss again and the corresponding gradient is used to actually update the network's parameters with respect to the parameters *before* the temporary update. This training process simulates domain shift and aids with generalization to unseen domains.

Although their method outperformed all other methods they compared to (four DG methods, among which were two regularization methods, one meta-learning and one augmentation) with 6-fold cross-validation, this was mostly a result of task-specific loss functions which do not generalize to tasks outside (prostate) segmentation. They also measured the performance of their method with plain meta-learning and showed that it performs similarly to an elaborate data augmentation scheme proposed in [28], which will be discussed in detail in the next section.

III. B. Increasing data diversity

This section describes the second method of achieving DG, which is increasing data diversity, with a total of ten papers discussed. This is often achieved using either augmentations (five papers) or by creating synthetic images (five papers). Synthetic images are generally used to generate either a wide variety of realistic MR images or to create images with an even larger variety that go beyond realistic MR contrasts. The former uses signal equations and quantitative maps (T_1, T_2, ρ) to generate data (two papers), whereas the latter uses segmentations of anatomical structures (three papers). A brief overview of these ten methods is given in Table II.

TABLE II
OVERVIEW OF METHODS IN THIS SUBSECTION, WHAT ANATOMY THEY STUDIED, ACROSS WHICH DOMAINS THEY TRIED TO GENERALIZE AND WHAT METHOD THEY USED FOR THIS

Ref.	Anatomy	Domains	Domain generalization method
[25]	Heart	Population and scanner	Augmentation
[28]	Heart and prostate	Scanner and sequence	
[29]	Breast	Scanner and sequence	
[30]	Heart	Population	
[31]	Heart and prostate	Scanner and sequence	
[32]	Brain	Scanner and sequence	Synthesis (quantitative maps)
[33]	Brain	Sequence	
[34]	Brain and heart	Scanner and sequence	Synthesis (segmentations)
[35]	Brain	Scanner and sequence	
[36]	Brain	Population, scanner and sequence	

Note that all papers focused on training a network for image segmentation, with the exception of [35], which focused on registration.

1) Augmentations

All five methods in this section use a set of basic augmentations from the possibilities shown in Table III. Here, brightness (B) refers to linear transformations of the image intensity and the gamma transformation (G) is a transformation that changes the image contrast by exponentiating each intensity by a random number γ, giving a non-linear intensity transformation (after normalization to the range 0-1). On top of these basic augmentations, additional (advanced) augmentations are also used in the last three papers of this section.

TABLE III
OVERVIEW OF THE TYPES OF AUGMENTATION USED IN THIS SUBSECTION PER REFERENCE: ROTATION (R), FLIPPING (F), SCALING (S), NON-LINEAR DEFORMATION (ND), GAUSSIAN NOISE (GN), BRIGHTNESS (B), GAMMA TRANSFORMATION (G), SHARPENING/BLURRING (S/B) AND ADDITIONAL AUGMENTATIONS.

Ref.	R	F	S	ND	GN	B	G	S/B	Additional
[25]	✓	✓	✓	✓	✓	✓	✓		
[28]	✓	✓	✓	✓	✓	✓	✓	✓	
[29]	✓		✓						✓
[30]	✓	✓	✓	✓		✓			✓
[31]	✓			✓	✓	✓	✓		✓

The authors of [25] devised an augmentation method for the segmentation of heart structures such that they could generalize to different scanners, populations and sequences. They placed first in the previously mentioned heart segmentation challenge [24]. Their method is relatively simple, because the authors argue that the domain shift resulting from differences in scanners, populations and sequences in cardiac MR images is actually rather small compared to what is seen in other areas of deep learning, where a network has to recognize e.g. a dog from a painting as well as from a picture. The augmentations they used are shown in Table III.

A large number of augmentations were also used in [28], which used these to train a network for whole prostate segmentation and a network for left atrial segmentation with generalization in mind. The augmentations were relatively similar to those used in [25] as can be seen in Table III. Four different domains were considered in [28] for the prostate scans and three for the heart scans, with the domains spanning multiple types of scanners and sequences. One of these domains was used to train a segmentation network for each of the tasks, while the rest were used for testing.

To compare their method they also trained a CycleGAN to harmonize images to the source domain that the network was trained on, which was always one domain out of four/three, for the prostate and heart datasets, respectively. They found that on both left atrial- and prostate segmentation, their method significantly outperformed CycleGAN-based harmonization. This is promising, because the CycleGAN-based approach requires training on data from the unseen domain, which is not always possible, whereas the data augmentation strategy does not. Augmentations are also easier to implement, because they are often part of standard DL libraries and do not require a second harmonization network and training/hyperparameter tuning of this second network. This is also the only example of a direct quantitative comparison of DG versus DH that was found in this review, showing that DG performs better in this case.

In [29] a smaller set of augmentations was used than the previous two methods (see Table III), but here the authors tested two different intensity transformations to generalize a whole-breast segmentation network trained on T1w images to unseen T2w images. On top of simple spatial augmentations consisting of rotations and scaling, they used intensity remapping and style-based augmentation. The first technique remapped each pixel value to a random value using a remapping function that consisted of a linear function with added random noise. The second technique used a pretrained style transfer network, which takes in an image and a style embedding to transfer any style to an input image. The network embeds the image domain as the style in this case. The style transfer network is also able to embed the domain of the input image, which was mixed with a random domain embedding and subsequently fed into the network to apply a random domain to the images, while not straying too far from the original image domain.

By training the network on T1w data augmented with either intensity remapping or style augmentation the authors achieved a performance on the T2w data that was comparable to training and evaluating on the T2w data. Both intensity augmentations performed similarly, but since style transfer is much slower to apply on the fly and harder to implement, intensity remapping is probably preferable. The authors did mention that the intensity remapping function required a lot of finetuning to get right, so a gamma transformation might be preferable as used in [25], [28], because this is also an intensity remapping, but it requires less finetuning.

So far the previous three methods have focused on randomly selected augmentations, which may or may not challenge the network on a given iteration. The authors of [30] used a more advanced augmentation method to train a network for the segmentation of the left ventricular myocardium from cardiac MR images of healthy subjects that could generalize to MR images of patient groups with four different pathologies (e.g. dilated cardiomyopathy and abnormal right ventricle). To do so, they employed an augmentation step designed to create challenging examples for the network.

More specifically, the authors focused on adversarial data augmentation, which is a method of creating augmentations that strongly perturb the output of the network and often comes in the form of additive noise. Instead of this local perturbation with random noise, the authors proposed the use of more global adversarial bias fields, which simulate the smoothly varying bias fields often seen in MR images due to magnetic field inhomogeneities. The bias fields were constructed such that they created a large distance between the predicted output and the newly predicted output after bias field multiplication by maximizing the Kullback-Leibler divergence (KL divergence) between these two outputs.

The authors of [30] used adversarial additive noise combined with random augmentations similar to those of [25], [28] (see Table III) as a baseline along with mixup + random augmentations and just random augmentations. They trained their network on data from healthy subjects and tested the

network on data from four populations with different cardiac pathologies using these different augmentation methods. The data of all five populations was obtained with similar sequences using two different Siemens scanners operating at 1.5 T and 3 T. They found that overall their method generalized better to the four pathological populations than the other three augmentation methods, while performing better than or comparably to the other methods on the training domain of healthy subjects.

The final method discussed in this section used advanced random augmentations and compares their method to that of [30] and also some of the basic random augmentations used in [25], [28] to train a cardiac segmentation network and a prostate segmentation network [31].

The first component of their method is a non-linear intensity transformation achieved by applying a shallow CNN initialized with random weights to the data. A random linear interpolation between the output of this CNN and the original image was subsequently constructed to form the augmented image. The second component is the removal of spurious correlations used by the network for its predictions. An example of this could be a network trained for the segmentation of the kidneys, which makes use of the fact that the spleen has a similar intensity as the kidneys and is always located next to the left kidney to determine which kidney is which. This spurious correlation could break in the presence of a strong bias field, pathologies in either the left kidney or spleen or a different type of sequence/scanner, making it an undesirable feature to learn for the network from a generalization standpoint.

The non-linear intensity transformation in and of itself is not enough to remove the spurious correlation in this example, because the intensities are quite similar and the transformation is spatially invariant. Therefore, the second component is added, which relies on the creation of two augmented images with the non-linear intensity transformation and subsequently randomly mixes these two images in a spatially varying manner to create two different images. The entire workflow is given schematically in Fig. 4. To enforce the network to learn domain invariant features a KL divergence term between the two predicted segmentations was used together with a segmentation loss.

To test the generalization capabilities of their method, the authors trained a network for cardiac segmentation on data acquired using a balanced steady-state free precession (bSSFP) sequence and applied it to data acquired using a late gadolinium enhanced bSSFP sequence (with different sequence parameters) and they trained a network for the segmentation of the prostate on data from one scanner and sequence and applied it to data from five other sites using different scanners and sequences. They compared against six other methods for domain generalisation, among which were random augmentations (see Table III) and random augmentations + adversarial bias fields as proposed in [30]. Note that random augmentations were applied in all six methods and their own.

On both the cardiac and prostate segmentation tasks their method outperformed all six other DG methods for the segmentation of three cardiac structures and whole-prostate, respectively in terms of DSC. However, they saw a decrease in DSC using their method compared to random augmentations on the training domain for the prostate dataset. This indicates that the better generalization capabilities sacrificed some performance on the training domain.

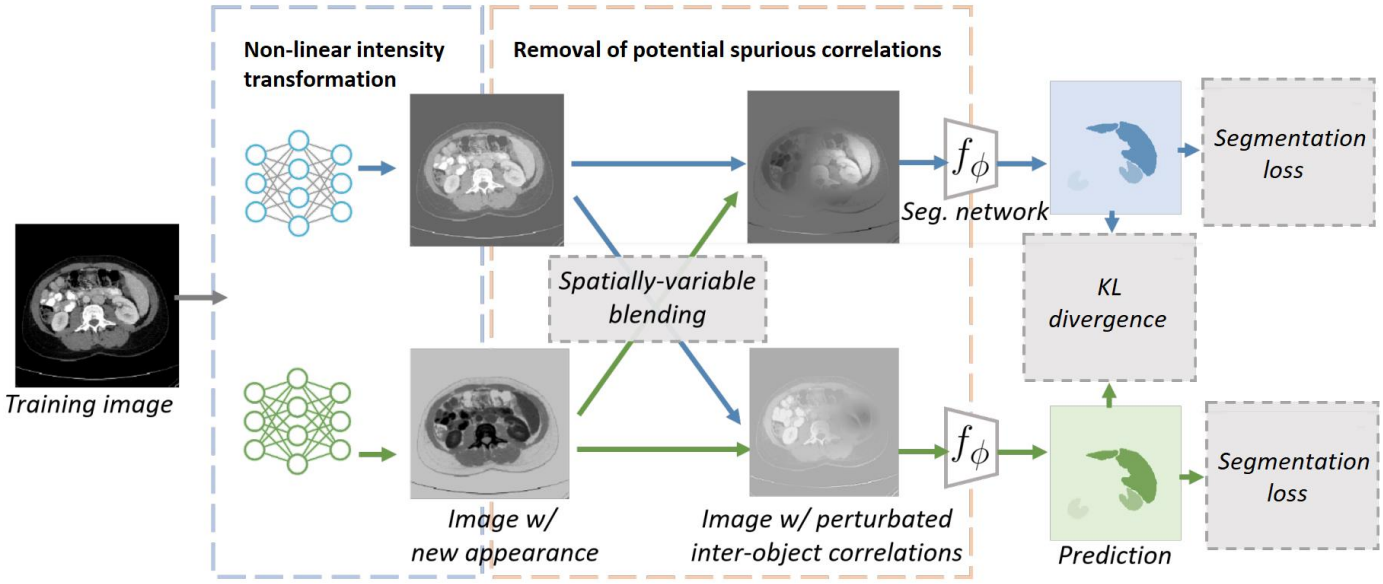

Fig. 4. Overview of the augmentation approach from [31], which takes in a single image and applies two different non-linear intensity transformations, the results of which are subsequently mixed in a spatially varying manner before being fed to the segmentation network. Adapted from [31].

2) Quantitative-map-based synthesis

The three methods in this subsection create highly varied data like in data augmentation methods, but using MR signal equations instead, simulating a wide variety of possible contrasts acquired during a normal acquisition.

In [32] quantitative maps (T_1, T_2, ρ) were used to simulate three different sequence types with the corresponding signal equations to generalize a brain segmentation network to scans from different scanners/sequences. The network was tasked with segmenting 13 brain structures. Each training batch contained a real T1w MPRAGE and synthetic T1w SPGR, T2w 3D Turbo Spin Echo and T1w MPRAGE to make the network robust to these three sequence types. Because their segmentation dataset did not contain quantitative maps, these were estimated from the T1w MPRAGE scans using three CNN's trained for T1-, T2- and ρ-map synthesis on a separate dataset containing T1w MPRAGE scans and quantitative maps.

Their method performed comparably or better than two non-DL methods (SAMSEG and multi-atlas registration and label fusion) on segmentation of the brain structures from real T1w SPGR data. They also showed that their method had the lowest variation in nine out of the 13 brain structure volumes across four different scans of 13 subjects created using different sequences and scanners.

The authors of [33] (a direct follow-up to their older method [37], which they outperformed) used a similar strategy to train a white- and grey matter segmentation network that could generalize to unseen domains of different sequences. Instead of predicting quantitative maps using CNN's they used real quantitative maps. Additionally, the authors provided the network with the signal equation parameters (e.g. repetition time, flip angle) to condition its output on the type of simulated input image.

They trained networks with simulated images that were similar to what was expected in the testing data by constraining for example the repetition- and echo times to a certain range. They showed that their method performed better on slightly different scans than a CNN trained on a single contrast, but also noted that the injection of signal equation parameters into the network did not give much of a performance boost.

3) Segmentation-based synthesis

The three methods discussed in this section all came from the same research group and made use of synthetic images to create diverse data for robustness to a wide range of inputs/domain shifts. Synthetic images were created using segmentations and this technique was first proposed in [38] for joint super resolution and image synthesis from arbitrary contrasts. This subsection focuses on the three follow-up papers, which included comparisons to other (DG) methods, in contrast to [38]. These three methods used similar synthesis strategies to train a generalizable network to:

- segment brain- and cardiac structures [34],
- register brain images [35] and
- perform skull-stripping (whole-brain segmentation) [36].

Before discussing their results a short overview of the synthesis strategy of these three papers is given, illustrated in Fig. 5. First, high resolution segmentations (created from real MR images) of the relevant anatomy are sampled from a small training set and spatially transformed with an affine and non-linear transformation followed by nearest-neighbour interpolation (Fig. 5a, 5b). Random intensities are then sampled for each of the segmentation labels using a Gaussian Mixture Model (GMM) conditioned on the label (Fig. 5c). The image is subsequently multiplied with a simulated bias field followed by a gamma transformation (Fig. 5d). To make the networks robust to images of different resolutions, a slice spacing and slice thickness are simulated using blurring and downsampling, followed by an upsampling back to the resolution of the original segmentations (Fig. 5e, 5f).

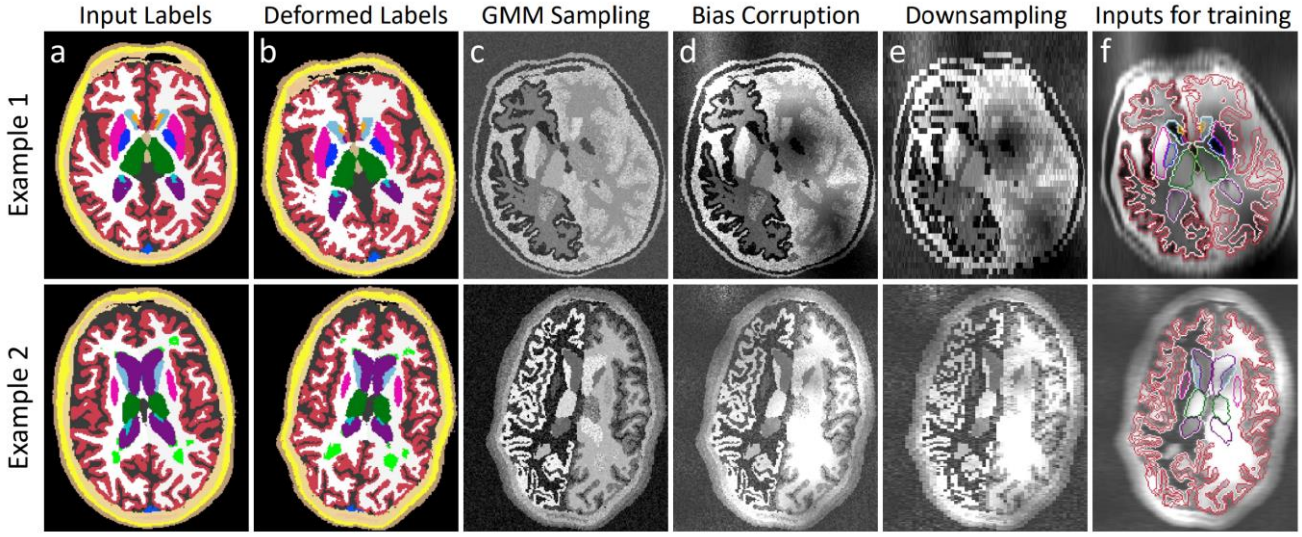

Fig. 5. Overview of the synthetic image creation of [34]–[36]. Adapted from [34].

Notice that the random spatial transformation, simulated bias field and gamma transformation are also often used as augmentations (see Table III). The difference between synthesizing images versus augmenting existing images are

two-fold in the case of segmentation: 1) because the images are synthesized from the segmentation maps, the images and corresponding labels are always completely aligned as opposed to augmented images, for which the correspondence between the images and labels depend on the method used for segmentation (e.g. labelled by a radiologist) and will almost always be lower. 2) The image intensities in the augmentation of existing data are more restricted than those of the synthetic images even with the use of gamma transformations and bias field simulation.

The methods of this section performed domain randomisation. which entails training a network on a far larger variety of images than would be encountered in reality. Domain randomisation ensures that the training data encompasses this smaller subset of realistic images, effectively making the network resilient to domain shifts larger than realistically possible for e.g. MR images [34]. These large domain shifts are particularly useful for segmentation, where global shape is important in determining which class a tissue belongs to, because CNN's typically focus on local structure and texture/intensity instead of global shape [39]. By varying these local features using random intensities, intensity transformations and non-linear spatial transformations the network is forced to learn more global shape features, because these are mostly preserved.

In [34] domain randomisation/data synthesis was used to train a network for the segmentation of brain structures. The network was evaluated on seven MRI domains spanning multiple sequences (T1w, T2w, proton density-weighted, Fluid-Attenuated Inversion Recovery) and scanners. Although the trained network had never seen real MRI data, it significantly outperformed two domain adaptation methods (test-time adaptation and SIFA; not discussed in this review) trained on a single source domain of T1w images in terms of DSC and 95% surface distance (SD95) on six of the seven MRI domains. It also outperformed a state-of-the-art non-DL segmentation method, SAMSEG on these six domains in terms of DSC and SD95.

The authors of [34] applied the same approach using segmentations of cardiac MR images instead to show that their method is not just limited to brain images, but they did not compare to other methods for this second task. They achieved DSC's of around 0.86 on both an MRI dataset and a CT dataset with seven labelled structures and comparable performance between the two datasets.

Although the segmentation map-based synthesis approach is most obviously applicable to segmentation tasks, it is also useable for learning image registration as shown in [35], where it was used to train a network for the registration of brain image pairs. The segmentations were used to measure the overlap between the images after application of the predicted transformation by the network using a DSC loss.

Brain segmentation maps were used for training, significantly outperforming two DL-based approaches using the same network (VoxelMorph [40] trained with either normalized mutual information or normalized cross correlation as the image similarity loss) in terms of DSC and surface distance on the domain that these networks were trained on and even more so on the domains that these networks were not trained on. Furthermore, their network managed to outperform two non-DL registration methods, NiftyReg and ANTS. They thereby showed that learning (global) shape features is not only beneficial for segmentation, but also for registration.

Finally, the authors applied roughly the same approach as the previous two papers in [36] for the learning of skull-stripping/whole-brain segmentation. They compared their method to five non-DL-based skull-stripping methods (e.g. ROBEX, BET, 3dSkullStrip) and one DL-based method. The non-DL-based methods were chosen based on popularity (citations) and effectiveness as shown in previous literature [36]. The DL-based method was the only top cited DL-based method that agreed to share their network, with others not making their network available.

The authors of [36] evaluated their method and the other methods on 15 MRI datasets spanning 15 sequences and multiple scanners. The mean surface distance (MSD) between the predicted and ground truth brain segmentation was calculated for all 15 datasets for the six baseline methods, resulting in 90 MSD values. By comparing the MSD's achieved by their own method to these values for each dataset, this gave 90 total comparisons between their method and the baselines. Out of these, their method significantly outperformed the baseline methods in 87 out of 90 comparisons in terms of MSD. Furthermore, their method achieved the highest DSC *and* lowest MSD on 80% of all test images, followed by 10% for the second best method (BET).

IV. Discussion

In this review an overview of five DH- and sixteen DG methods has been given, which aimed to counteract the adverse effects of domain shifts in MRI data on the performance of DL-based approaches. An overview of all 21 methods is given in the appendix in Table A1. The next subsections first discuss the pros and cons of these methods starting with a high-level overview and delving into the individual methods later on. Afterwards, the strengths and limitations of the papers in general and of this review are discussed, followed by suggestions for future research.

IV. A. Pros and cons

DH-based and DG-based methods approach domain shifts in different ways and are therefore applicable in different scenarios. DH maps data from different domains to one domain to remove domain-specific information, whereas DG trains networks to be robust to input coming from different domains.

DH can be used in two ways in the space of DL. It can be used to harmonize data to the training domain of a trained task network such that this network can still be applied to new data from different domains, or it can be used to harmonize data to a reference domain such that a task network can be trained on the harmonized data. In the second case, the act of creating domain-independent features and actually learning the task are split up across the harmonization- and task networks.

Because DH is often performed using neural networks, the same considerations with regards to domain shift have to be taken into account. If a harmonization network is trained to harmonize T1w scans from different scanners and populations to a reference domain of one type of scanner and population, it

will not generalize to T2w scans for example. Therefore, DH is mostly applicable in the case where 1) data from all the domains that we wish to harmonize are available or 2) the domain shifts between the data that the harmonization network was trained on and that it will be later applied to for harmonization are small.

If larger domain shifts are expected to occur during deployment of the network, the harmonization network itself will need to be robust to domain shifts. This could be achieved using DG as in e.g. [14], [15], which used domain-adversarial learning. The question then is if it is better to train a harmonization network using DG and train a task network on harmonized data or if it is better to train a task network directly with DG. Conceptually, it is easier to split the process up into removing domain-specific information and learning a task using a harmonization network and task network, respectively. On the other hand, training a task network with DG aims to remove domain-specific information and learn a task at the same time.

Although DH approaches allow for the splitting of these two seemingly separate objectives, it runs into two issues. Firstly, this type of approach forces one to choose a particular reference domain to harmonize all other domains towards. The question is how this reference domain should be chosen and if this is actually gives the optimal representation of the underlying anatomy. Secondly, the task network is at the mercy of the harmonization network. If the harmonization network removes important features or adds features that should not be there during harmonization, there is no way for the task network to reverse this error, leading to possible errors in the final output. This is especially true when using GANs, which have a higher risk of this type of behaviour [17]. This might be part of the reason why CycleGAN-based harmonization was outperformed by augmentation strategies in [28] for both heart- and prostate segmentation generalization, even though the CycleGAN-based harmonization had the benefit of training on the test domain, which was not the case for the augmentation approach.

Training a task network directly with DG solves both of the aforementioned problems in theory. The first problem is solved by allowing a network to create its own internal domain-independent representation of the data, which will be optimized directly using the task-specific loss function. This internal representation is probably more optimal than simply choosing a reference domain from the available domains in the training data, because it is not restricted to a relatively small set of representations as in the DH case. The second problem is alleviated because the task network performs both the task and removal of domain-specific information itself, no longer relying on another network's harmonization performance.

In conclusion, DH approaches are mostly applicable if no domain shifts are expected during deployment of the harmonization network or if they are expected to be small. Outside of DL-based methods, however, DH could also be useful for harmonization towards domains which certain non-DL approaches work better on, because these can also suffer from performance loss under domain shift, albeit to a smaller degree than DL-based approaches [36]. Finally, DH has the benefit of not requiring any labelled data/retraining for the downstream task. In the case that a trained task network already exists, but there is no (or not enough) labelled data to retrain this network such that it generalizes to new unseen domains, DH could be used to still be able to apply this network to new domains. In most other cases, such as when large domain shifts are expected during deployment or the extent of the domain shifts is unknown, training a task network with DG directly would be more practical as discussed before. Therefore, DG is preferable over DH if the option of training a network for a task from scratch exists. This is because DG is more applicable when the aim is to train a neural network that can be applied to data that is acquired at a later time without having to know what these domains will be or requiring data from these new domains.

1) Data harmonization

Section II. A and II. B reviewed GAN-based and encoder-decoder-based DH. In general, the use of image-level discriminators seems to be disadvantageous, because these risk feature hallucination [17]. The disadvantage of image-level discriminators as used in GANs was shown indirectly in [14], which used an encoder-decoder-based method with an adversarial loss on the latent space, outperforming CycleGAN in terms of image similarity after harmonization. More direct evidence of the disadvantage of using image-level discriminators was shown in [15], which showed that cycling from one domain to another and back to the original domain gave a much larger quantitative and qualitative difference in image similarity when an image-level discriminator was used during training, indicating the removal/addition of features.

Therefore, non-GAN approaches should arguably be preferred over GAN-based approaches. Of these, [14] and [15] have the most potential, because [14] allows for finetuning on unseen domains in an unsupervised fashion, whereas [15]'s method showed that zero-shot harmonization of an unseen domain was possible, albeit from a similar domain to the training domains.

2) Domain generalization

As the DG section spans a large range of papers, this subsection first focuses on those that provide quantitative comparisons with other methods to draw conclusions. The authors of [31] showed that their advanced augmentations with random non-linear intensity transformations and removal of spurious correlations (combined with basic augmentations, see Table III for details) outperformed adversarial bias field augmentation [30] (also combined with basic augmentations) and pure basic augmentations, showing the strength of their approach in the augmentation-based DG methods. Because some of these methods that the authors of [31] compared to were also used as comparisons in other DG papers, this allows for the comparison of these papers to the three aforementioned methods as well, albeit across different tasks.

The authors of [30] showed that their method of creating adversarial bias fields (+ basic augmentations) outperformed mixup (+ basic augmentations) for the generalization of heart segmentation to unseen populations. This allows us to connect their findings to those of [20], which compared domain adversarial learning to mixup for the generalization of knee segmentation to unseen populations, both combined with a small set of basic augmentations (spatial transformations, gamma transformation and smoothing). In [20] domain

adversarial learning and mixup were found to perform equally well, even though the test domain was similar to one of the training domains on which the adversarial learning was performed. One caveat with this finding is that the authors of [20] trained both the domain discriminator and task network jointly from the start, which might have impeded performance.

By training the domain discriminator and task network jointly immediately, the latter is provided with noisy gradients, because the domain discriminator has not learned to discriminate the domains yet as the latent representation of the task network is still mostly random noise, leading to unstable training. Instead, a more optimal training scheme should have been used, like in [18], [19], [21], to achieve a better performance. The authors of [18], [19], [21] first trained the task network separately from the domain discriminator (optionally also separately training the domain discriminator as in [19], [21]) with no adversarial loss maximization for the task network until a certain point (often a fixed number of epochs) after which both networks were trained jointly.

Although training in this more stable manner might have improved the performance of domain adversarial training in [20], the fact still stands that this approach had access to an unlabelled dataset that was similar to the test set during training, but performed similarly to mixup, which did not have this advantage.

Furthermore, the authors of [30] showed that mixup + basic augmentations and basic augmentations performed similarly, which allows us to compare these two against meta-learning as used in [27]. The authors of [27] showed that meta-learning and basic augmentations as proposed in [28] performed similarly, indicating that basic augmentations, mixup + basic augmentations and meta-learning might offer similar performance boosts. The main difference between meta-learning versus the other two methods is that meta-learning requires multiple domains to be available during training to simulate domain shift, whereas augmentations/mixup can be applied to data from a single domain and still give good generalization as demonstrated in [28].

It is not possible to definitively conclude which method is better than which, because they were used across different tasks (all segmentation, but different domains, anatomies and basic augmentations). However, assuming that the results across the different tasks generalize to other tasks, the following ranking from best to worst DG method can be constructed:

1. Advanced augmentations with removal of spurious correlations + basic augmentations [31]
2. Advanced augmentations with adversarial bias fields + basic augmentations [30]
3. Mixup + basic augmentations [20],
   domain adversarial learning + basic augmentations [20],
   basic augmentations [28],
   meta-learning [27].

Domain adversarial learning and meta-learning have a large dependence on the data used for training and the magnitude of domain shifts inside the training data. For example, a network trained on datasets of T1w scans from different scanners with domain adversarial learning/meta-learning will not generalize well to T2w data, because the domain shift is much larger than what is seen during training. These two methods might therefore work well in the case where the training data contains domain shifts equal to/larger than what is expected to be encountered during deployment of the network, for example when a network is trained to be applied to T1w scans with similar sequences from different scanners only.

In the case where the network needs to be robust to a larger range of inputs, the augmentation methods seem more appropriate as they allow the network to become robust to a wider variety of images than could be acquired by real scanners. This might also be the reason why the advanced augmentation methods seem to perform better than e.g. domain-adversarial- and meta-learning. This same principle holds for the synthetic images based on segmentations, which we argue are preferable to those constructed using quantitative maps. [34]–[36] show the large benefit of creating a wide range of images, which is much harder to achieve using quantitative maps, because this would entail the use of many carefully chosen signal equations and sequence parameters to cover all bases. Note that it is important to cover a large variety of inputs, such that the possible MR contrasts are a subset of the inputs seen during training. If the synthetic/augmented data is not varied enough, this is not guaranteed and could lead to worse generalization.

As discussed before, the segmentation-based synthesis allows for the creation of more varied data than the augmentation-based methods—with the possible exception of [31]—while guaranteeing a one-to-one correspondence between the segmentation and input image, which might give this method an edge over the augmentation-based methods.

IV. B. Limitations

All data augmentation/synthesis methods that have been reviewed were applied to segmentation specifically, with the exception of [35]. As mentioned before, the large variety of data forces the network to focus more on global shapes than local shape, texture and intensities. This is intuitively useful for segmentation (and registration), because global shape should remain relatively unchanged across different domains. The question remains if this more shape-focused learning is also beneficial for e.g. classification and synthesis or for the segmentation of small structures that are only present in a small number of voxels.

Additionally, some pathologies might only be clearly visible using certain image contrasts. In that case, should e.g. a classification network really be generalizable to a large number of contrasts? It could be argued that generalizability to a small range of contrasts that adequately show the pathology (and generalizability to all scanners) would be more beneficial in this case. However, just because the testing data will probably be restricted to a small set of image contrasts, this might not be necessary during training as demonstrated in [34]–[36]. It could also be the case that training on a wider variety of training data using advanced augmentations/data synthesis might allow the network to better capture relevant features. The trained network could then be applied to those scans that show the pathology well.

As for quantitative comparisons between methods, there were not many papers that compared more than two DH/DG

methods all at once. More results could have been found by searching for e.g. “multi-site MRI challenge”, which are valuable because these challenges often withhold some of the data from one or multiple sites (often with different sequences and scanners per site) from the participants for testing purposes, giving a way of quantitatively comparing multiple methods of countering domain shifts on a single challenge. These challenges might have been missed with the keywords “data harmonization”, “domain generalization” or “domain shift” if these were not explicitly mentioned in the challenge paper. However, as in [24] challenges are not always strict enough about the solutions that can be sent in, leading to large variations in approaches beyond the methods used for robustness to domain shifts.

IV. C. Future research

For future research ideally more restrictive challenges should be organized, where the participants are provided with e.g. a baseline network architecture and loss function to be used with the restriction that components can only be added to improve robustness to domain shift, e.g. data augmentation or an adversarial loss component and domain discriminator for domain-adversarial learning.

The role of augmentation/synthesis should also be studied more in-depth for tasks other than segmentation/registration. For augmentation methods this would be straightforward, but especially for the segmentation-based synthesis this becomes a bit more difficult. The latter shows a lot of potential in extracting domain-independent features and could be interesting to explore for tasks other than segmentation, because its training data is of high quality due to the built-in one-to-one correspondence between input- and target output data. One way this approach might be used for e.g. classification is to first train a network for segmentation as in [34]–[36], leading to a network that gives domain-independent features in the final feature maps. The trained network could then be frozen and used as a feature extractor, supplying the final feature maps and optionally the segmentation map to a second smaller network (as most of the feature extraction has been performed already), which could then be trained for classification.

Finally, an interesting property of the methods trained with heavy data augmentation or the segmentation-based synthetic data is that they can be applied to a wide variety of contrasts. Because multiple contrasts are acquired in most MRI examinations in the clinic, these networks could be applied to all of the images acquired in a single examination and give a more robust prediction.

Because not all contrasts highlight certain pathologies of interest as well, a weighting would have to be given to each of the acquired images to take this into account when combining the images. This could be a heuristic weighting or it could be informed by e.g. a second neural network, tasked with mixing the outputs to match a final target output, conditioned on the types of inputs used.

V. Conclusion

In this review we have discussed DH and DG methods, concluding that DH and DG are both viable if the expected domain shifts during deployment of a neural network are small. In the more likely case of unknown/larger expected domain shifts, DG seems to be the more practical option. Increasing the diversity of the training data beyond realistic MR contrasts seems to be the most promising direction in DG using either augmentations or synthetic data. Finally, if a trained task network already exists and data from an unseen domain is acquired without labels, DH gives the option of harmonizing these data to the training domain such that the trained task network can be applied without retraining it on the new domain. Therefore, both DH and DG have their own unique use cases.

References

[1] A. S. Lundervold and A. Lundervold, “An overview of deep learning in medical imaging focusing on MRI,” *Z. Med. Phys.*, vol. 29, no. 2, pp. 102–127, May 2019, doi: 10.1016/J.ZEMEDI.2018.11.002.

[2] V. M. Bashyam *et al.*, “Deep Generative Medical Image Harmonization for Improving Cross-Site Generalization in Deep Learning Predictors,” *J. Magn. Reson. Imaging*, vol. 55, no. 3, pp. 908–916, Mar. 2022, doi: 10.1002/JMRI.27908.

[3] B. E. Dewey *et al.*, “Deep harmonization of inconsistent mr data for consistent volume segmentation,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11037 LNCS, pp. 20–30, 2018, doi: 10.1007/978-3-030-00536-8_3/FIGURES/5.

[4] M. Ghafoorian *et al.*, “Transfer learning for domain adaptation in MRI: Application in brain lesion segmentation,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10435 LNCS, pp. 516–524, 2017, doi: 10.1007/978-3-319-66179-7_59/FIGURES/3.

[5] B. E. Dewey *et al.*, “DeepHarmony: A deep learning approach to contrast harmonization across scanner changes,” *Magn. Reson. Imaging*, vol. 64, pp. 160–170, Dec. 2019, doi: 10.1016/J.MRI.2019.05.041.

[6] Y. He, A. Carass, L. Zuo, B. E. Dewey, and J. L. Prince, “Autoencoder based self-supervised test-time adaptation for medical image analysis,” *Med. Image Anal.*, vol. 72, p. 102136, Aug. 2021, doi: 10.1016/J.MEDIA.2021.102136.

[7] H. Takao, N. Hayashi, and K. Ohtomo, “Effect of scanner in longitudinal studies of brain volume changes,” *J. Magn. Reson. Imaging*, vol. 34, no. 2, pp. 438–444, Aug. 2011, doi: 10.1002/JMRI.22636.

[8] R. T. Shinohara *et al.*, “Volumetric Analysis from a Harmonized Multisite Brain MRI Study of a Single Subject with Multiple Sclerosis,” *AJNR Am. J. Neuroradiol.*, vol. 38, no. 8, p. 1501, Aug. 2017, doi: 10.3174/AJNR.A5254.

[9] I. J. Goodfellow *et al.*, “Generative Adversarial Networks,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11046 LNCS, no. NeurIPS, pp. 1–9, Jun. 2014, doi: 10.48550/arxiv.1406.2661.

[10] J. Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks,” *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2017-October, pp. 2242–2251, Dec. 2017, doi: 10.1109/ICCV.2017.244.

[11] Y. Liu *et al.*, “A 3D Fully Convolutional Neural Network With Top-Down Attention-Guided Refinement for Accurate and Robust Automatic Segmentation of Amygdala and Its Subnuclei,” *Front. Neurosci.*, vol. 14, p. 260, May 2020, doi: 10.3389/FNINS.2020.00260/BIBTEX.

[12] Y. Choi, Y. Uh, J. Yoo, and J. W. Ha, “StarGAN v2: Diverse Image Synthesis for Multiple Domains,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 8185–8194, Dec. 2019, doi: 10.48550/arxiv.1912.01865.

[13] B. E. Dewey *et al.*, “A Disentangled Latent Space for Cross-Site MRI Harmonization,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12267 LNCS, pp. 720–729, 2020, doi: 10.1007/978-3-030-59728-3_70/FIGURES/6.

[14] L. Zuo *et al.*, “Information-Based Disentangled Representation Learning for Unsupervised MR Harmonization,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12729 LNCS, pp. 346–359, 2021, doi: 10.1007/978-3-030-78191-0_27/TABLES/4.

[15] K. Fatania *et al.*, "Harmonisation of scanner-dependent contrast variations in magnetic resonance imaging for radiation oncology, using style-blind auto-encoders," *Phys. Imaging Radiat. Oncol.*, vol. 22, pp. 115–122, Apr. 2022, doi: 10.1016/J.PHRO.2022.05.005.

[16] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, May 2015, vol. 9351, pp. 234–241, doi: 10.1007/978-3-319-24574-4_28.

[17] J. P. Cohen, M. Luck, and S. Honari, "Distribution Matching Losses Can Hallucinate Features in Medical Image Translation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11070 LNCS, pp. 529–536, Sep. 2018, doi: 10.1007/978-3-030-00928-1_60.

[18] H. Guan, Y. Liu, E. Yang, P. T. Yap, D. Shen, and M. Liu, "Multi-site MRI harmonization via attention-guided deep domain adaptation for brain disorder identification," *Med. Image Anal.*, vol. 71, p. 102076, Jul. 2021, doi: 10.1016/J.MEDIA.2021.102076.

[19] K. Kamnitsas *et al.*, "Unsupervised domain adaptation in brain lesion segmentation with adversarial networks," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10265 LNCS, pp. 597–609, 2017, doi: 10.1007/978-3-319-59050-9_47/FIGURES/4.

[20] E. Panfilov, A. Tiulpin, S. Klein, M. T. Nieminen, and S. Saarakkala, "Improving Robustness of Deep Learning Based Knee MRI Segmentation: Mixup and Adversarial Domain Adaptation," *Proc. - 2019 Int. Conf. Comput. Vis. Work. ICCVW 2019*, pp. 450–459, Aug. 2019, doi: 10.48550/arxiv.1908.04126.

[21] C. M. Scannell, A. Chiribiri, and M. Veta, "Domain-Adversarial Learning for Multi-Centre, Multi-Vendor, and Multi-Disease Cardiac MR Image Segmentation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12592 LNCS, pp. 228–237, 2021, doi: 10.1007/978-3-030-68107-4_23/TABLES/1.

[22] J. Corral Acero, V. Sundaresan, N. Dinsdale, V. Grau, and M. Jenkinson, "A 2-Step Deep Learning Method with Domain Adaptation for Multi-Centre, Multi-Vendor and Multi-Disease Cardiac Magnetic Resonance Segmentation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12592 LNCS, pp. 196–207, 2021, doi: 10.1007/978-3-030-68107-4_20/FIGURES/4.

[23] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond Empirical Risk Minimization," *6th Int. Conf. Learn. Represent. ICLR 2018 - Conf. Track Proc.*, Oct. 2017, doi: 10.48550/arxiv.1710.09412.

[24] V. M. Campello *et al.*, "Multi-Centre, Multi-Vendor and Multi-Disease Cardiac Segmentation: The MMs Challenge," *IEEE Trans. Med. Imaging*, vol. 40, no. 12, pp. 3543–3554, Dec. 2021, doi: 10.1109/TMI.2021.3090082.

[25] P. M. Full, F. Isensee, P. F. Jäger, and K. Maier-Hein, "Studying Robustness of Semantic Segmentation Under Domain Shift in Cardiac MRI," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12592 LNCS, pp. 238–249, 2021, doi: 10.1007/978-3-030-68107-4_24/TABLES/6.

[26] C. Finn, P. Abbeel, and S. Levine, "Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks," *34th Int. Conf. Mach. Learn. ICML 2017*, vol. 3, pp. 1856–1868, Mar. 2017, doi: 10.48550/arxiv.1703.03400.

[27] Q. Liu, Q. Dou, and P. A. Heng, "Shape-Aware Meta-learning for Generalizing Prostate MRI Segmentation to Unseen Domains," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12262 LNCS, pp. 475–485, 2020, doi: 10.1007/978-3-030-59713-9_46/FIGURES/3.

[28] L. Zhang *et al.*, "Generalizing Deep Learning for Medical Image Segmentation to Unseen Domains via Deep Stacked Transformation," *IEEE Trans. Med. Imaging*, vol. 39, no. 7, pp. 2531–2540, Jul. 2020, doi: 10.1109/TMI.2020.2973595.

[29] L. S. Hesse, G. Kuling, M. Veta, and A. L. Martel, "Intensity augmentation for domain transfer of whole breast segmentation in MRI," Sep. 2019, doi: 10.48550/arxiv.1909.02642.

[30] C. Chen *et al.*, "Realistic Adversarial Data Augmentation for MR Image Segmentation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12261 LNCS, pp. 667–677, Jun. 2020, doi: 10.48550/arxiv.2006.13322.

[31] C. Ouyang *et al.*, "Causality-inspired Single-source Domain Generalization for Medical Image Segmentation," Nov. 2021, doi: 10.48550/arxiv.2111.12525.

[32] A. Jog, A. Hoopes, D. N. Greve, K. Van Leemput, and B. Fischl, "PSACNN: Pulse Sequence Adaptive Fast Whole Brain Segmentation," *Neuroimage*, vol. 199, pp. 553–569, Jan. 2019, doi: 10.48550/arxiv.1901.05992.

[33] P. Borges *et al.*, "Acquisition-invariant brain MRI segmentation with informative uncertainties," Nov. 2021, doi: 10.48550/arxiv.2111.04094.

[34] B. Billot *et al.*, "SynthSeg: Domain Randomisation for Segmentation of Brain Scans of any Contrast and Resolution," Jul. 2021, doi: 10.48550/arxiv.2107.09559.

[35] M. Hoffmann, B. Billot, D. N. Greve, J. E. Iglesias, B. Fischl, and A. V. Dalca, "SynthMorph: Learning Contrast-Invariant Registration Without Acquired Images," *IEEE Trans. Med. Imaging*, vol. 41, no. 3, pp. 543–558, Mar. 2022, doi: 10.1109/TMI.2021.3116879.

[36] A. Hoopes, J. S. Mora, A. V. Dalca, B. Fischl, and M. Hoffmann, "SynthStrip: Skull-Stripping for Any Brain Image," Mar. 2022, doi: 10.48550/arxiv.2203.09974.

[37] P. Borges *et al.*, "Physics-informed brain MRI segmentation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11827 LNCS, pp. 100–109, 2019, doi: 10.1007/978-3-030-32778-1_11/FIGURES/4.

[38] J. E. Iglesias *et al.*, "Joint super-resolution and synthesis of 1 mm isotropic MP-RAGE volumes from clinical MRI exams with scans of different orientation, resolution and contrast," *Neuroimage*, vol. 237, Dec. 2020, doi: 10.48550/arxiv.2012.13340.

[39] N. Baker, H. Lu, G. Erlikhman, and P. J. Kellman, "Deep convolutional networks do not classify based on global object shape," *PLOS Comput. Biol.*, vol. 14, no. 12, p. e1006613, Dec. 2018, doi: 10.1371/JOURNAL.PCBI.1006613.

[40] G. Balakrishnan, A. Zhao, M. R. Sabuncu, J. Guttag, and A. V. Dalca, "VoxelMorph: A Learning Framework for Deformable Medical Image Registration," *IEEE Trans. Med. Imaging*, vol. 38, no. 8, pp. 1788–1800, Sep. 2018, doi: 10.1109/TMI.2019.2897538.

[41] X. Huang, M. Y. Liu, S. Belongie, and J. Kautz, "Multimodal Unsupervised Image-to-Image Translation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11207 LNCS, pp. 179–196, Apr. 2018, doi: 10.48550/arxiv.1804.04732.

[42] L. Yu, X. Yang, H. Chen, J. Qin, and P. A. Heng, "Volumetric ConvNets with Mixed Residual Connections for Automated Prostate Segmentation from 3D MR Images," *Proc. AAAI Conf. Artif. Intell.*, vol. 31, no. 1, pp. 66–72, Feb. 2017, doi: 10.1609/AAAI.V31I1.10510.

[43] F. Isensee, P. F. Jaeger, S. A. A. Kohl, J. Petersen, and K. H. Maier-Hein, "nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation," *Nat. Methods 2020 182*, vol. 18, no. 2, pp. 203–211, Dec. 2020, doi: 10.1038/s41592-020-01008-z.

[44] S. Liu *et al.*, "3D Anisotropic Hybrid Network: Transferring Convolutional Features from 2D Images to 3D Anisotropic Volumes," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11071 LNCS, pp. 851–858, Nov. 2017, doi: 10.48550/arxiv.1711.08580.

Appendix

TABLE A1

OVERVIEW OF ALL DATA HARMONIZATION/DOMAIN GENERALIZATION PAPERS DISCUSSED, WHICH METHOD THEY USED, WHAT ANATOMY THEY STUDIED, WHICH TASK THEY EVALUATED THE GENERALIZATION CAPABILITIES WITH, ACROSS WHICH DOMAINS THEY TRIED TO GENERALIZE AND WHICH NETWORK ARCHITECTURE THEY USED FOR THIS. IN THE CASE OF DATA HARMONIZATION, THE HARMONIZATION NETWORK ARCHITECTURE IS GIVEN.

Ref.	Method	Anatomy	Task	Domains	Network
[2]	DH: GAN-based	Brain	Age estimation	Scanner and sequence	StarGAN v2 [12]
[11]		Brain	Amygdala segmentation	Population, scanner and sequence	CycleGAN [10]
[13]	DH: encoder-decoder-based	Brain	N.A.	Population, scanner and sequence	U-Net [16]
[14]		Brain	N.A.	Scanner and sequence	U-Net
[15]		Brain	Radiomic features	Scanner	MUNIT [41]
[18]	DG: domain-independent feature learning	Brain	Disorder identification and disease progression prediction	Population, scanner and sequence	Own design
[19]		Brain	Brain lesion segmentation	Scanner and sequence	Own design
[20]		Knee	Femoral and tibial cartilage tissue segmentation	Population	U-Net
[21]		Heart	Left and right ventricle cavities and left ventricle myocardium segmentation	Population and scanner	U-Net
[22]					
[27]		Prostate	Whole prostate segmentation	Scanner and sequence	Mix-residual-U-Net [42]
[25]	DG: augmentation	Heart	Left and right ventricular cavities and left ventricular myocardium segmentation	Population and scanner	nnU-Net [43]
[28]		Heart and prostate	Left atrial and prostate segmentation	Scanner and sequence	AH-Net [44]
[29]		Breast	Whole breast segmentation	Scanner and sequence	U-Net
[30]		Heart	Left ventricular myocardium segmentation	Population	U-Net
[31]		Heart and prostate	Left and right ventricle and myocardium	Scanner and sequence	U-Net
[32]	DG: quantitative-map-based synthesis	Brain	Segmentation of thirteen brain structures	Scanner and sequence	U-Net
[33]		Brain	White- and grey matter segmentation	Sequence	nnU-Net
[34]	DG: segmentation-based synthesis	Brain and heart	Segmentation of fourteen brain- and seven heart structures	Scanner and sequence	U-Net
[35]		Brain	Registration	Scanner and sequence	U-Net
[36]		Brain	Whole brain segmentation	Population, scanner and sequence	U-Net