Medical Image Analysis 87 (2023) 102825

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

Unpaired, unsupervised domain adaptation assumes your domains are already similar

Gijs van Tulder [a,b,*], Marleen de Bruijne [b,c]

[a] Data Science group, Faculty of Science, Radboud University, Postbus 9010, 6500 GL Nijmegen, The Netherlands
[b] Biomedical Imaging Group, Erasmus MC, Postbus 2040, 3000 CA Rotterdam, The Netherlands
[c] Department of Computer Science, University of Copenhagen, Universitetsparken 1, 2100 Copenhagen, Denmark

ARTICLE INFO

Keywords:
Domain adversarial learning
Domain adaptation
Representation learning
Transfer learning

ABSTRACT

Unsupervised domain adaptation is a popular method in medical image analysis, but it can be tricky to make it work: without labels to link the domains, domains must be matched using feature distributions. If there is no additional information, this often leaves a choice between multiple possibilities to map the data that may be equally likely but not equally correct. In this paper we explore the fundamental problems that may arise in unsupervised domain adaptation, and discuss conditions that might still make it work. Focusing on medical image analysis, we argue that images from different domains may have similar class balance, similar intensities, similar spatial structure, or similar textures. We demonstrate how these implicit conditions can affect domain adaptation performance in experiments with synthetic data, MNIST digits, and medical images. We observe that practical success of unsupervised domain adaptation relies on existing similarities in the data, and is anything but guaranteed in the general case. Understanding these implicit assumptions is a key step in identifying potential problems in domain adaptation and improving the reliability of the results.

1. Introduction

Modern deep learning methods for medical image analysis achieve impressive results, but the models they produce often generalize poorly to data from different scanners or different medical centers. This is especially inconvenient in medical imaging because it can be time-consuming and expensive to obtain the ground-truth annotations for a new training set. Domain adaptation methods address this problem by adapting models trained on data from one domain, the *source*, to data from another, the *target*. If the domain adaptation step works well, models trained for existing datasets can be applied to data from new domains with only a limited performance loss. Similarly, domain adaptation can be used to combine data from multiple sources in a single model, either by modeling the differences between domains or by reducing them.

1.1. Unsupervised domain adaptation

Domain adaptation comes in many shapes and forms (see Guan and Liu, 2021, for a recent overview of applications in medical imaging). In this paper we study *unsupervised domain adaptation*, which assumes that labeled data is only available for the source domain. Some methods for unsupervised domain adaptation learn the translation between domains from paired data, such as scans of the same patient in different scanners. Here, we investigate a more challenging setting: unsupervised domain adaptation without paired samples.

Without information on individual sample pairs, the mapping between domains must be learned on a distribution level. To do this, a common assumption is that although the data from the source and target domains *looks* different, the *underlying structure and tissue types* are quite similar. For example, a brain scan might look different in different scanners, but the anatomical information is the same. This correspondence can be exploited to learn a mapping between domains: if the domains have similar underlying structure and tissue types, we should expect the features and outputs to have a similar distribution as well.

1.2. Image-to-image translation

Many unsupervised domain adaptation methods are based on *image-to-image translation*: by translating images from the target domain to the source domain, they can be analyzed using the existing classifiers trained on source data. For example, the popular CycleGAN model (Zhu et al., 2017) is optimized using a cycle-consistency loss, which minimizes the reconstruction loss of a source–target–source translation, and

* Corresponding author at: Biomedical Imaging Group, Erasmus MC, Postbus 2040, 3000 CA Rotterdam, The Netherlands.
E-mail addresses: g.vantulder@cs.ru.nl (G. van Tulder), marleen.debruijne@erasmusmc.nl (M. de Bruijne).

https://doi.org/10.1016/j.media.2023.102825
Received 29 April 2022; Received in revised form 30 March 2023; Accepted 17 April 2023
Available online 23 April 2023

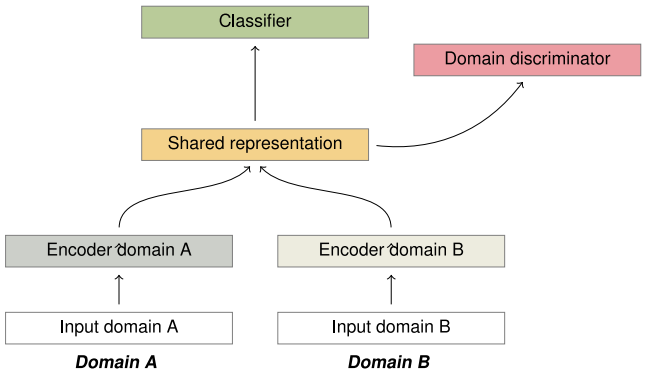


Fig. 1. The adversarial domain adaptation model uses a separate encoding branch for each domain. The output of these encoders is forwarded to a shared classification network and to a domain discriminator. A domain adversarial learning objective is applied to encourage the encoders to learn a shared, domain-invariant representation space.

an adversarial loss that discriminates between real and translated target images. An alternative approach for image-to-image translation uses style transfer separate content and appearance information (Yang et al., 2019; Chen et al., 2020b,b, 2021b).

Image-to-image translation is complex and relatively inefficient. The translation model must translate all information in the images, but only some of that information is useful to the subsequent classification or segmentation model. Moreover, the focus on reconstruction loss may remove useful information that is difficult to translate between images. In many cases, finding a perfect translation might be impossible. For example, a translation between MRI and CT may only preserve information that is captured by both modalities.

1.3. Learning domain-invariant representations

An alternative to image-to-image translation is *domain adaptation in feature space* by learning domain-invariant representations. After mapping domain-specific inputs to a common, domain-invariant feature representation, the same classifier or segmentation model can be used for all domains. If the dataset contains paired samples, the domain-specific mappings can be learned with a loss that compares the representation of the same sample across domains. Without paired samples, the mappings can be learned by aligning the feature distributions for both domains, e.g., using a distribution similarity metric such as the Maximum Mean Discrepancy loss (MMD, Gretton et al., 2008), with a variational autoencoder (Wu and Zhuang, 2021), or with Optimal Transport (Ackaouy et al., 2020; Al Chanti and Mateus, 2021).

In this paper, we use the popular approach of *domain adversarial learning* (Fig. 1) (Ganin et al., 2017). This method relies on a domain discriminator that is trained to predict the domain of a sample given its feature representation. By using this discriminator in an adversarial learning objective for the feature encoding model, the encoder is encouraged to learn domain-invariant representations. Tzeng et al. (2017) describe a general framework for adversarial discriminative domain adaptation (ADDA) that covers many variants of this approach. Kamnitsas et al. (2017) present an early application of domain adversarial learning to brain lesion segmentation.

We investigate the application of representation learning to unsupervised domain adaptation with unpaired samples, where we assume that labels are only available for the source domain and there is no direct link between samples in the source and target domains. We use domain adversarial learning to implement our domain adaptation objective, but we believe that many of our conclusions also hold for other methods.

1.4. Why does this even work?

Domain adversarial learning is a popular method in medical image analysis (Guan and Liu, 2021), often with good results, but there has been relatively little research into *why* it works. At first glance, domain adversarial learning makes very few assumptions about the data, and should be able to align any pair of domains just by matching their feature distributions. In practice, we argue in this paper, aligning distributions is not sufficient: there is usually more than one way to match the domains, which means that additional assumptions about the data are needed to find the correct solution.

In early work on this topic, Ben-David S. Blitzer et al. (2010a) and Ben-David S. Luu et al. (2010b) explored the theoretical bounds of the error of a domain adaptation model (Ben-David S. Blitzer et al., 2010a) and discussed the assumptions for a successful domain adaptation result (Ben-David S. Luu et al., 2010b). Most importantly, they suggest that the unlabeled source and target distributions should be similar. More recently, Zhao et al. (2019) provided a theoretical analysis of domain adaptation by learning invariant representations, i.e., intermediate features which have a similar distribution in the source and target domains. Zhao et al. (2019) show that in general, learning an invariant representation and achieving a small error on the source domain is not sufficient to guarantee a small error on the target domain, because the labeling function may be different for both domains.

In this paper, we explore these themes from a medical imaging perspective. We hypothesize that a successful domain adaptation using adversarial learning requires explicit or implicit assumptions about the data, or more specifically: assumptions about the similarities between domains. We explore what these assumptions can be, and show why they help to obtain useful domain adaptation results. We investigate a number of data and model characteristics that often appear in medical imaging and that might explain why medical domain adversarial learning is successful. We explore these properties in several practical experiments, comparing results for datasets with different properties and different network architectures. We conclude that identifying these implicit biases is a key step in obtaining reliable domain adaptation results.

1.5. Outline

Section 2 presents an overview of related work in adversarial domain adaptation for medical images. Section 3 describes the unsupervised domain adaptation approach. Section 4 discusses the problems with this approach, and why it should not work in theory. Section 5 explains why it sometimes does work in practice. Section 6 introduces the metrics used to evaluate the results. Section 7.1 describes the technical implementation of the experiments. Section 7.2 shows the experiments on a synthetic dataset, followed by Section 7.3 on MNIST digits and Section 7.4 on two medical datasets. Sections 8 and 9 provide a discussion and conclusion.

2. Related work

We summarize the main trends on adversarial domain adaptation in a medical context. We discuss two approaches: image-level domain adaptation, which translates images between domains, and feature-level domain adaptation, the approach used in this paper, which learns domain-invariant feature representations. Guan and Liu (2021) provide a recent survey of domain adaptation in medical imaging, covering adversarial learning and other methods.

2.1. Image-level domain adaptation with a cycle-consistency loss

Many adversarial domain adaptation works use image-to-image translation with a cycle-consistency loss, based on the CycleGAN model (Zhu et al., 2017). Cohen et al. (2018) point out that this type of image-to-image translation may not be ideal. They argue that distribution matching is sensitive to differences in the sample distribution between the source and target domains, which can lead to unrealistic and incorrect translations. They illustrate this with a CycleGAN model that adds spurious tumor patterns when translating between brain MRI protocols. The CyCADA model (Hoffman et al., 2018) adds a semantic consistency loss that aligns the translated image on a feature level or on a task-specific level, such as the output of a classification model.

In medical imaging, the CycleGAN approach has been used for MRI-to-CT image synthesis (Wolterink et al., 2017; Yang et al., 2018; Zhou et al., 2021), multi-contrast MRI (Dar et al., 2019), fundus imaging (Ju et al., 2021), chest X-ray (Li et al., 2020a), histopathology (de Bel et al., 2021), and ultrasound (Zhou et al., 2021) images. The basic cycle-consistency loss is sometimes extended with additional, application-specific constraints, e.g., by encouraging structural or anatomical consistency between domains (Cai et al., 2019; Chen et al., 2021a; Jiao et al., 2020). Other works using CycleGAN align domains based on the output of auxiliary tasks such as segmentation (Ren et al., 2021; Tomar et al., 2021; Tomczak et al., 2021), or by directly matching feature values (Chen et al., 2020a; Yu et al., 2020). Some other works use atlas registration (Gao et al., 2019) or a student–teacher model with inter- and intra-domain teachers (Li et al., 2020b.) to improve the results.

In general, the CycleGAN approach alone is not sufficient to learn a reliable translation (Hoffman et al., 2018). Additional constraints, and corresponding assumptions about the domains, are required to get usable results. Recent publications show promising results with image-to-image translation methods based on style transfer, as an approach to separate content and appearance information. For example, Yang et al. (2019) propose image-to-image translation with disentangled representations, linking domains both on feature and image levels. Chen et al. (2020b) use feature disentanglement to combine shape priors and image appearance. Chen et al. (2021b) report good results by encoding anatomical information separately from appearance information. The utility of spatial similarities between domains is also recognized by Wang and Zheng (2022), who observe that cross-domain image translation can be improved by including a semantic segmentation task. A recent challenge on unsupervised domain adaptation for cardiac MRI segmentation (Zhuang et al., 2022) also reports good results for style-transfer-based image-to-image translation methods. Like CycleGAN, style transfer-based models usually assume that images in different domains to have a similar spatial arrangement.

2.2. Feature-level domain adaptation

Adversarial domain adaptation by learning domain-invariant feature representations (Tzeng et al., 2017), without explicitly reconstructing images from the target domain, is also commonly used for medical image classification and segmentation. Kamnitsas et al. (2017) presented an early version of this approach for brain lesion segmentation. The method was later applied for many other tasks, such as anatomical structure segmentation (Bian et al., 2020), multi-modal brain MRI (Guan et al., 2021), colonoscopy images (Liu et al., 2021), or fundus imaging (Shen et al., 2020). Instead of learning a fully domain-invariant model, some approaches try to disentangle domain-invariant and domain-specific features (Hu et al., 2020; Pei et al., 2021), which allows them to exploit domain-specific information where necessary.

Feature-level domain adaptation can be extended with additional constraints, e.g., by adding structural constraints on the output of a segmentation model. Bateson et al. (2021) argue that adversarial training may not be suitable for adapting segmentation networks, and suggest using domain-invariant prior knowledge about common anatomical structures to direct the adaptation. Similarly, Cui et al. (2021) used several structural constraints to capture common cardiac structure across MRI and CT. More indirectly, Wang et al. (2019) applied an adversarial domain discriminator to a segmentation output. Li et al. (2020b.) provided additional semantic feature maps to the discriminator, to exploit domain-invariant spatial patterns.

Domain adaptation can also be guided by adding auxiliary tasks to the learning objective. For example, Koohbanani et al. (2021) used domain-specific pretext tasks in a self-supervision setup. Luo et al. (2020) used task-specific discriminators to improve domain invariance. Chen et al. (2019) proposed a combination of feature-level and image-level methods.

As an alternative to adversarial matching of feature distributions, some approaches minimize the distance between class and feature distributions of across domains using metrics such as the Kullback–Leibler divergence or mutual information. For example, Bateson et al. (2020) use this method to include a learned class-prior to match distributions in a class-sensitive way. Liang et al. (2020) propose a classification-based domain adaptation approach to obtain a similar class distribution between domains.

3. Methods

3.1. Domain adaptation with a neural network

In this paper, we consider domain adaptation in a deep neural network with the following architecture: an encoder that maps the domain-specific input to a latent, domain-invariant feature representation, and a shared prediction model that uses the intermediate representation to make a prediction. We use classification as the prediction task in this paper, but this could also be a segmentation or regression task. The domain adaptation in the encoder can take two forms: using a single encoder that is used for both domains, or using a separate, domain-specific encoder for each domain.

The first approach requires a single, common model that works well for data from both domains. Since it uses the same feature extraction path for both domains, it will automatically map both domains to the same representation if the domains are fairly similar. However, the approach provides limited flexibility to adapt to larger differences between domains, and is likely to focus on domain-invariant features that have similar appearance in both domains.

The second approach uses a separate encoding path for each domain. We use this architecture in this paper. In contrast to a shared encoder, domain-specific encoders can accommodate large differences between domains: if the encoders are complex enough, they can map the inputs to a shared encoding that is common to both domains. However, the increased power and flexibility also increase the risk that the encoders learn inconsistent mappings, since there are no shared features that link the two encoding branches. We will revisit this limitation in Section 4.

3.2. Adversarial domain adaptation

The source encoder and the shared prediction model can be trained with a supervised learning objective, computed on labeled data from the source domain. To train the target encoder and learn a domain-invariant feature representation, we need an unsupervised objective based on the unlabeled target data and data from the source domain. In this paper, we use an adversarial domain adaptation objective.

Adversarial learning (Goodfellow et al., 2014) is commonly used to train generative models. A discriminator is trained to discriminate between samples from a real distribution and samples generated by a generator model. By optimizing the generator to maximize the loss of the discriminator, the samples generated by the model will start to resemble those from the real distribution.

In domain adversarial learning (Ganin and Lempitsky, 2015; Tzeng et al., 2017), the discriminator is presented with feature representations of samples from the source and target domain, and is trained to predict the domain of each sample. The discriminator loss is included as an adversarial term in the learning objective for the encoders, which encourages them to learn domain-invariant representations that have similar distributions in both domains.

3.3. Architecture and learning objectives

Fig. 1 shows the model with domain-specific encoders as it is used in this paper. We denote the domain-specific encoders as F_{src} for the source and F_{tgt} for the target domain. Given an input $\mathbf{x}$, we use the appropriate encoder $F \in \{F_{\text{src}}, F_{\text{tgt}}\}$ to compute the representation $F(\mathbf{x})$. This representation is then used as input for a shared classification model G to compute the prediction $\hat{y} = G(F(\mathbf{x}))$.

The learning objective consists of a classification component and a domain-adversarial component. The classification component is computed only for the source samples, using the ground-truth label y to compute the binary cross-entropy loss:

$$\mathcal{L}_{\text{class}} = -y \log \hat{y} - (1-y) \log (1-\hat{y}). \tag{1}$$

A separate domain discriminator D is used to encourage the two encoders to produce domain-invariant representations. The discriminator is trained with a binary cross-entropy loss to predict the domain of a sample given its intermediate feature representation:

$$\mathcal{L}_{\text{disc}} = \begin{cases} -\log D\left(F_{\text{src}}(\mathbf{x})\right), & \text{if } \mathbf{x} \text{ is from the source domain;} \\ -\log\left(1 - D\left(F_{\text{tgt}}(\mathbf{x})\right)\right), & \text{if } \mathbf{x} \text{ is from the target domain.} \end{cases} \tag{2}$$

We reuse this learning objective $\mathcal{L}_{\text{disc}}$ as an adversarial term in the learning objective for the encoder.

During training, we optimize the encoder and classifier weights to minimize the classification loss and maximize the discriminator loss:

$$\mathcal{L}_{\text{combined}} = \lambda_{\text{class}} \mathcal{L}_{\text{class}} - \lambda_{\text{disc}} \mathcal{L}_{\text{disc}}. \tag{3}$$

4. Problem analysis

In the absence of paired samples, the domain adaptation model can only compare domains at a distribution level. This has consequences for the quality and correctness of the results.

4.1. Two phases of domain adaptation

For the following analysis, we will divide the unsupervised domain adaptation task in two phases. First, the method must determine the structure of the input space for each domain, e.g., by identifying clusters of samples. Second, the method must match the structures in both domains in order to map the feature representations of samples from one domain to the other. If both phases are successful, the domain adaptation will result in the correct classification on the target domain.

Our analysis is further based on the assumption that domain adaptation learns a smooth mapping between domains: samples that are close together in the target domain will most likely be mapped close together in the source domain.

For simplicity, for this problem analysis, we will assume that the samples in each domain can be grouped in a number of distinct clusters. In practice, we may not be able to find perfectly distinct clusters in the data – for example, because samples from different classes may have very similar appearance and classes may overlap – but this will not affect our general conclusion.

Ideally, each class would correspond to a single cluster in each domain, and the task of domain adaptation would be to link each cluster to the correct cluster in the other domains. In practice, it is likely that the classes are more heterogeneous and consist of multiple subclusters. This complicates the task of the domain adaptation algorithm, which must now identify all subclusters and link them to the correct classes in the other domain.

Both domain adaptation phases must be successful to obtain a good classification result. Observing the target classification accuracy at the end is not sufficient to identify which of the two parts failed: a low target accuracy combined with a high source accuracy could mean that both clustering and mapping failed, but it could also mean that the model found the right clusters but mapped them incorrectly between domains.

4.2. Unsupervised domain adaptation requires additional assumptions

Consider a thought experiment with a balanced binary classification problem, in which each class contains fairly homogeneous samples. Given the in-class homogeneity, it is easy to find the correct clusters. Linking those clusters across domains is more difficult: without additional information, it is impossible to say which cluster in the target domain belongs to which cluster in the source domain. As a result, domain adaptation has only a 50% chance of success. In unsuccessful cases the clustering may still work, while the classification accuracy may be close to zero because the clusters are linked incorrectly.

Observe that the problem in this simple example would not occur if the classes were not balanced. Provided that the imbalance was similar for both domains, the model could use the size of each cluster to learn a correct mapping.

However, the result also depends on the assumption that the samples within each class are sufficiently homogeneous. In practice, this will almost never be the case. For example, in some applications different types of tissue might map to the same class. In segmentation tasks, voxels near the edge of a structure may have a different appearance from voxels located in the center, even if the whole structure belongs to a single class, and the representation of near-edge voxels may even vary with orientation. For this analysis, we therefore assume that each class consists of multiple subclusters that are internally homogeneous. This makes it more difficult to find the correct solution, since the required class balance can be achieved with different combinations of subclusters.

Consider an experiment in which the data is subdivided in 10 homogeneous subclusters of equal size. If the class balance in the source domain is 80–20, that is, 8 and 2 subclusters per class, this can be replicated in the target domain by mapping any combination of two subclusters to the minority class. Since there is no way for the algorithm to identify which combination is correct, the domain adaption is likely to fail even if it discovers the clusters correctly.

In this paper, we argue that the conclusions for these thought experiments can be extended to domain adaptation on real datasets. We provide experimental verification of these specific results on synthetic data in Section 7.2.

5. Exploiting domain-invariant properties

In the previous section, we argued that unsupervised domain adaptation is unlikely to learn correct mappings if there is no information to link subclusters across domains. In practice, of course, this is too pessimistic. Unlike the dataset in our example, most real-world datasets will have some domain-invariant properties that can be exploited to align domains.

The outcome of adversarial domain adaptation depends on the initial representations, which usually depend on randomly initialized weights. Since the training makes small, incremental changes to the encoders to match distributions, it can increase similarity of clusters that are already similar, but it is unlikely to swap entire clusters. If the initial guess was correct, the final mapping is likely to be correct as well.

Fortunately, the initial mapping and subsequent optimization are not completely random, but depend on biases in the data and the

model. If these biases are helpful, domain adaptation is more likely to succeed. In this section, we introduce four domain-invariant properties that are commonly seen in medical imaging data and may provide a useful source of domain adaptation bias. We will then discuss how these properties can influence the domain adaptation results implicitly.

5.1. Similar class imbalance

In Section 4.2, we argued that class imbalance might be used to link domains with homogeneous classes. Many real-life datasets show some class imbalance, but most are also heterogeneous. Our thought experiment showed that this makes the imbalance less useful, because the subclusters in the data can be combined in arbitrary ways to obtain the required class balance. The experiments later in this paper confirm this.

5.2. Similar intensities

If the average image intensities are consistent between domains, e.g., if a class that is brighter in one domain is also brighter in the other domain, this similarity can be used to learn the correct mapping between domains. This assumption often holds for images from the same imaging modality. For example, CT images from different scanners will have roughly similar intensity patterns.

This similarity can be exploited explicitly (models with shared encoders are based on this assumption), but it can also affect the domain adaptation implicitly. Here, we argue that the architecture and initialization of the model can interact with intensity similarities in the data to bias the model towards particular mappings. Given a random initialization of the weights and a standard activation function, the magnitude of the input intensities is reflected in the representation: on average, a class with inputs around zero will produce smaller absolute feature values in the encoder output than a class with larger input values. This initial bias is consistent for all domain-specific encoders, and can be used to map classes with similar intensity to similar feature values.

5.3. Similar spatial structures

In applications with spatial inputs, source and target domains may have similar spatial arrangements. For example, in MRI and CT images of the same anatomy, the modalities produce images that show the same anatomical structures, even if the appearance is different. We argue that domain adaptation could exploit spatial similarities like these if the models use convolution.

With convolutional encoders, the latent representation preserves the spatial structure of the input. Even with a random initialization of the weights, the output of convolutional encoders in different domains will generate representations that are spatially similar. As long as the classes have the same spatial arrangement in both domains, these similarities could be exploited by the model to link the domains, even if the structures themselves have a different appearance.

5.4. Similar local texture and intensity distributions

A fourth source of similarities is local texture. Especially in segmentation tasks, texture information could be used to identify components if the textures are similar across domains. Using convolution makes the encoders sensitive to type and amount of texture: heavily textured areas may produce a different convolution output than areas with a lighter texture, even with random initialization of the weights. This could bias the encoders to learn similar representations for similarly textured areas, which would lead to a correct mapping if the texture has similar meaning across domains. In medical imaging, this kind of texture similarity can appear in multi-view images from the same imaging modality, such as multi-modal MRI or smaller variations in scanning protocol. On the other hand, cross-modality applications such as MRI-to-CT could have different textures in each domain, which could lead to a bias towards incorrect mappings.

6. How to measure domain adaptation success?

We employ several metrics to measure the performance of the models, based on the two phases in the domain adaptation process that we identified in Section 4.1: finding clusters in each domain, and linking those clusters across domains.

6.1. Measuring the correctness of the mapping

Ultimately, the performance of domain adaptation is defined by the *classification accuracy* on the target domain. In the experiments in this paper, we compute the classification accuracy on the source domain and on the target domain. Since the classifier is trained only on the source domain, we expect the performance on the target domain to be lower, but ideally the two should be as close as possible.

6.2. Measuring mapping quality

However, as discussed in Section 4.1, classification accuracy alone does not provide the full picture, since it measures the combined success of both domain adaptation phases. We use three metrics to evaluate the clustering phase separately.

6.2.1. Compensated accuracy

A simple case of cross-domain confusion in a binary classification task is that the domain adaptation method correctly finds the two classes in the target domain, but maps them to the incorrect class in the source domain. To measure this effect, we define the *compensated accuracy* as $\max(\text{accuracy}, 100\% - \text{accuracy})$.

As an example, consider a binary classification task where for setting A each run has a target accuracy of 50%, and where for setting B each run has a random target accuracy of either 0% or 100%. For both settings, the mean accuracy over multiple runs would be 50%. However, the results are clearly not the same: in experiment A, the classification completely fails to identify the classes, whereas in experiment B, the classes are separated correctly, but are sometimes mapped incorrectly between the domains. The compensated accuracy discriminates between these two cases: the mean compensated accuracy of setting A is 50%, while the mean for setting B is 100%.

6.2.2. Mapping confidence

In more complicated problems with heterogeneous classes, we can assume that each class is made up of several subclusters. We define a domain adaptation *confidence* score that measures whether the domain adaptation model correctly identifies the subclusters in the data, independent of whether they are assigned to the correct class.

The metric is defined using subcluster labels. We first compute the subcluster confusion matrix CM and the class balance CB:

$$\mathrm{CM}(Y, C) = \sum_i I(\hat{y}_i = Y, c_i = C), \tag{4}$$

$$\mathrm{CB}(Y) = \frac{1}{N} \sum_i I(y_i = Y), \tag{5}$$

where $I(\cdot)$ is the indicator function, $Y \in 0, 1$ is a binary class, C is a subcluster, N is the number of samples, and $\hat{y}_i$, y_i, c_i are the predicted class, the ground-truth class, and subcluster of sample i, respectively. We then compute the class-balanced weighted confusion matrix WCM and the class difference CD:

$$\mathrm{WCM}(Y, C) = \mathrm{CM}(Y, C)/(2 \cdot \mathrm{CB}(Y)) \tag{6}$$

$$\mathrm{CD} = \sum_C \mathrm{WCM}(0, C) - \mathrm{WCM}(1, C). \tag{7}$$

Finally, we compute the confidence as

$$\text{Confidence} = \sum_C \max_Y(\mathrm{WCM}(Y, C)) - |\mathrm{CD}|\,. \tag{8}$$

The confidence score ranges between 0% and 100%. If the model identifies all subclusters correctly (i.e., samples from one subcluster are all assigned to the same class), the confidence score will be 100%, independent of the correctness of the classification. If the model achieves no clustering (e.g., samples from one subcluster are equally distributed over the two classes), the score is 0%. For a dataset with two homogeneous classes, the confidence is equal to the compensated accuracy.

Computing the confidence score requires a subcluster label for each sample in the target domain. Since these labels are not available in an unsupervised domain adaptation setting, this metric cannot be applied in practical applications, but we include it as a measure in our experiments to gain insight in the behavior of the algorithms.

6.2.3. Linear CKA

At the level of the encoder outputs, we compute the *representation similarity* using linear CKA (centered kernel alignment, Kornblith et al., 2019). Linear CKA measures the content-based feature similarity while allowing for differences in representation, giving an indication of how much information is shared by both domains. The method is often used to compare the feature representations of different networks trained on the same data, but we use it to compare representations of paired samples across domains. We refer to Kornblith et al. (2019) for the full definition. In our experiments, the linear CKA ranges from 0 (no alignment) to 100 (complete alignment).

7. Experiments and results

7.1. Implementation

We used neural networks to implement the domain-specific encoders F_{src} and F_{tgt}, the classifier G, and the domain discriminator D. The architectures of these networks are described in the following sections. In some experiments, we varied the level of the intermediate representation: we used the same set of layers for $F + G$ combined, but changed how they are divided between the encoders F and the classifier G (Fig. 3). The discriminator and classifier were optimized with a binary cross-entropy objective, using a gradient reversal layer between the discriminator and the encoders to implement the adversarial objective. All models were implemented in PyTorch[1] and trained using the Adam optimizer until convergence. Detailed architectures and hyperparameters are shown in Appendix.

7.2. Experiments with synthetic data

7.2.1. Data and architecture

We constructed a synthetic, binary classification problem with 10 input features, $\mathbf{x} \in \mathbb{R}^{10}$, and generated samples for two domains with identical or different input representations (Table 1), according to the following settings:

- For "Two $-1/+1$", we constructed a problem with two clusters: samples $[-1,-1,-1,-1,-1,-1,-1,-1,-1,-1]$ for class 0 and $[1,1,1,1,1,1,1,1,1,1]$ for class 1 in both domains (i.e., both domains received the same input).
- The variant "Two $-1/+1$, inverted" used the same type of samples, but we inverted the labels in the target domain: $[-1,\dots,-1]$ corresponded to class 1 and $[1,\dots,1]$ to class 0, simulating a very strong difference between domains.
- Similarly, "Two 0/1" and "Two 0/1, inverted" used samples with values $[0,\dots,0]$ and $[1,\dots,1]$ with equal or swapped classes, respectively.

Table 1
Synthetic datasets.
Datasets with two clusters used a feature vector filled with the same value. Datasets with ten clusters used a one-hot encoding with the feature corresponding to the cluster set to 1. Uniformly distributed noise was added to all features.

Synthetic dataset	Clusters	Source	Target
Two $-1/+1$	2	$-1/+1$	$-1/+1$
Two $-1/+1$, inverted	2	$-1/+1$	$+1/-1$
Two 0/1	2	0/1	0/1
Two 0/1, inverted	2	0/1	1/0
Ten	10	One-hot	One-hot

- Finally, "Ten" included samples with one-hot encoding, representing 10 different clusters: from $[1,0,0,0,0,0,0,0,0,0]$ to $[0,0,0,0,0,0,0,0,0,1]$. In our experiments, we assigned each cluster to one of the output classes, depending on the required class balance: for example, we assigned 2 clusters to class 0 and 8 clusters to class 1 to simulate a $20-80$ class balance.

For all settings, we added random noise to all features, sampled from a uniform $[-0.5, 0.5]$ distribution. This created many unique samples, without introducing class overlap. All experiments used the same, very simple architecture with linear encoders and decoders (Appendix, Fig. A.4).

7.2.2. Results

We ran the experiment described in Section 4.2 with the synthetic datasets. With homogeneous and balanced classes (experiment "Synthetic two $-1/+1$, Balanced 50–50" in Table 2), the model obtained a perfect classification accuracy on the source domain. On the target domain, however, the average classification accuracy was much lower. Looking closer, we observed that the target accuracy in individual runs was either 0% or 100%, while the compensated accuracy is always 100% for both domains. This confirmed our earlier prediction that the model would easily find the clusters in the data, but would be unable to reliably find the correct link between domains.

Next, we tried an experiment with unbalanced classes (experiments "Synthetic two $-1/+1$, Unbalanced 20–80" and "— 80–20"). The class imbalance helped the model to find the correct mapping, resulting in a perfect target accuracy in all runs. As hypothesized in Section 4.2, class imbalance was not sufficient in datasets with heterogeneous classes. When we performed the same experiment with heterogeneous classes (experiments "Synthetic ten"), we saw that the model failed to learn a good target classification. The high confidence scores indicate that the model was able to find the subclusters, but was unable to link them correctly between domains. We confirmed this by inspecting the confusion matrices (Appendix, Table A.6).

Finally, we found that the domain adaptation was sensitive to the representation of the input features. We repeated the experiments with homogeneous classes, but switched the input features from $\{-1,+1\}$ to $\{0,1\}$ (experiments "Synthetic two 0/1"). With these input values, even with balanced classes, the model learned a perfect accuracy on the target domain in almost all runs. We explain this surprising result with the bias predicted in Section 5.2: the representation of the data interacted with the model, introducing a bias that caused the model to learn the same representation for both domains. We found confirmation in the results for experiments with inverted target features (experiments "Synthetic 0/1, inverted"), in which the models reliably learned the incorrect mapping.

7.3. Experiments with MNIST digits

7.3.1. Data

We used the 28 × 28-pixel MNIST[2] digit images with intensities scaled to $[0,1]$, using the original training and test splits. We converted

[1] The source code for our experiments is available at https://vantulder.net/code/2023/uuda/.

[2] http://yann.lecun.com/exdb/mnist/

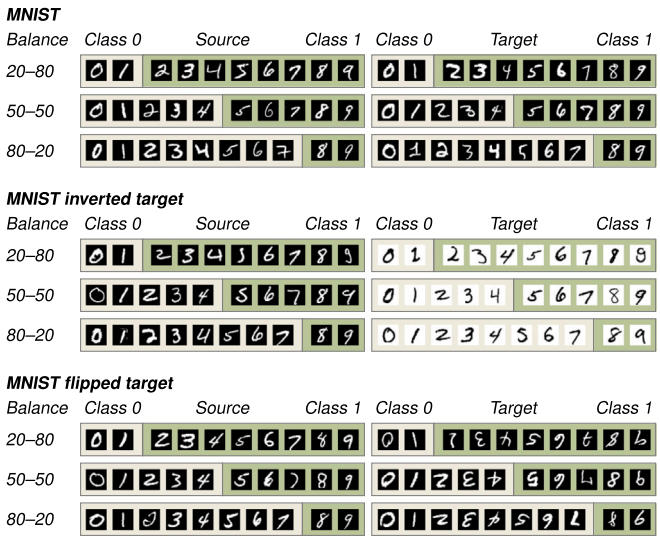


Fig. 2. MNIST datasets used in the experiments. The target domain is either the same as the source (MNIST), contains inverted images (MNIST inverted), or contains flipped images (MNIST flipped). The required class balance (20–80, 50–50, 80–20) is obtained by combining MNIST digits into two classes.

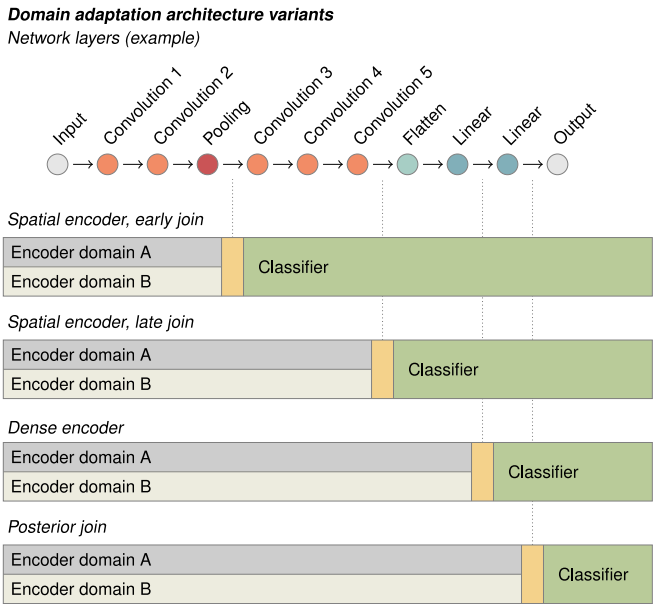


Fig. 3. Overview of the four architecture variants used in the experiments. For each architecture, the diagram shows the location of the shared representation (yellow) where the output of the two encoders is linked. The *spatial encoder* architectures join the two domains at a spatial representation level. The *dense encoder* architecture joins the two domains after the first fully connected layer. The *posterior join* architecture joins the domains just before the final activation function. The network shown here is an example: see the Appendix for the architecture in each experiment.

the original 10-class problem into a binary classification task by grouping the digits $\{0,1,2,3,4\}$ and $\{5,6,7,8,9\}$ (Fig. 2). To simulate a 20–80 or 80–20 class imbalance, we respectively classified $\{0,1\}$ vs $\{2,3,\ldots,9\}$ and $\{0,1,\ldots,7\}$ vs $\{8,9\}$. In all cases, we used the original digit labels to compute our subcluster-based confidence metric.

We performed experiments with three variations for the target domain (Fig. 2): 1. standard, with original images, similar to the source domain; 2. inverted, with inverted intensities (1−the original intensity) to remove intensity-based similarities; 3. flipped, with images horizontally and vertically mirrored to remove most of the spatial similarities between domains.

7.3.2. Architectures

We used a convolutional network with domain-specific encoders (Fig. 3, see the Appendix for full details: Fig. A.5). For the *spatial encoder* models, we joined the domain-specific encoders at the final spatial layer, just before global pooling. This gave the domain adaptation access to the final spatial feature maps. For the *dense encoder* models, we joined the encoders just after the global pooling layer, which meant that the domain adaptation method did not receive any spatial information.

7.3.3. Results

The results in Table 3 show that the domain adaptation model relied on spatial and intensity similarities to link the domains. In all experiments, the models with spatial encoders achieved a higher target accuracy than the models with dense encoders. The spatial encoders failed when they were applied to a data with a flipped target domain, because there were no spatial similarities to rely on. For digits that look similar when flipped (6 and 9), the similarities can even work to confuse the model further. At the same time, the models with spatial encoders were able to learn with large intensity shifts: the target accuracy on a target domain with inverted images was similar to that on standard images. However, the linear CKA scores were lower, which suggests that the representation still depended on intensity information.

The dense encoders had a low target accuracy on the standard domain with balanced classes, but showed a reasonably high confidence. This indicates that they could still find some clusters in the data. The models failed completely when the target images were inverted, which shows that they relied on intensity similarities to link the domains.

Surprisingly, the linear CKA scores of the dense encoder model dropped when the target images were flipped. This is counter-intuitive, because these models did not receive any spatial feature maps. We suspect this could be evidence for our fourth bias (Section 5.4): the early convolution layers encoded local texture information that influenced the later, global feature representations.

7.4. Experiments with brain MRI and cardiac CT/MRI

We present demonstrations on two medical imaging datasets: on brain MRI and on cardiac CT/MRI. We think it is likely that subclusters as discussed in this paper appear in any realistic dataset. However, to properly evaluate and observe the behavior, we required datasets with known subset labels. We created these datasets by combining multiple classes to create a binary classification task, where each class contains multiple subclasses, and then used the original class labels as the subclusters in our analysis.

7.4.1. Brain MRI dataset: BRATS

Our first demonstration uses brain MRI scans from the BRATS 2015 dataset (Menze et al., 2015). This brain tumor segmentation dataset includes four MRI sequences (T1, T1 with contrast, T2, FLAIR) and manual segmentations of four tumor components (necrosis, edema, non-enhancing tumor, and non-enhancing tumor). We extracted 2D patches of 15×15 pixels, labeled with the class of the center pixel and balanced to have an equal number of samples per class. We defined a binary classification problem by combining the BRATS labels into two classes: necrosis/edema and non-enhancing/enhancing tumor, which roughly corresponded to the outer and inner part of the segmentation, respectively. We used the original class labels as the subclusters in our analysis.

Table 2
Results for experiments with synthetic data, showing mean validation performance averaged over 25 models.
Models were trained to convergence. Sparkline plots show the distribution of results for individual runs.

	Accuracy (%)		Compensated accuracy (%)		Confidence	
	Source	Target	Source	Target	Source	Target
Synthetic two +1/ − 1						
Unbalanced 20–80	100.0	100.0	100.0	100.0	98.9	99.1
Balanced 50–50	100.0	64.0	100.0	100.0	99.1	99.3
Unbalanced 80–20	100.0	100.0	100.0	100.0	98.8	98.9
Synthetic two +1/ − 1, inverted target						
Unbalanced 20–80	100.0	100.0	100.0	100.0	99.1	99.1
Balanced 50–50	100.0	40.0	100.0	100.0	99.0	99.0
Unbalanced 80–20	100.0	100.0	100.0	100.0	98.8	99.0
Synthetic two 0/1						
Unbalanced 20–80	100.0	99.9	100.0	99.9	99.3	98.1
Balanced 50–50	100.0	92.0	100.0	100.0	99.4	99.2
Unbalanced 80–20	100.0	99.2	100.0	99.2	99.2	94.8
Synthetic two 0/1, inverted target						
Unbalanced 20–80	99.6	58.4	99.6	66.2	98.2	43.4
Balanced 50–50	100.0	4.0	100.0	100.0	99.2	99.0
Unbalanced 80–20	100.0	96.0	100.0	96.0	98.9	79.2
Synthetic ten						
Unbalanced 20–80	100.0	72.2	100.0	72.2	98.9	98.9
Balanced 50–50	100.0	47.6	100.0	65.3	99.3	98.1
Unbalanced 80–20	100.0	68.8	100.0	68.8	99.0	98.7

Table 3
Results for experiments with MNIST data, showing mean validation performance averaged over 25 models.
Models were trained to convergence. Sparkline plots show the distribution of results for individual runs.

	Accuracy (%)		Confidence		Linear CKA
	Source	Target	Source	Target	Target
MNIST, spatial encoder					
Unbalanced 20–80	99.6	98.2	97.9	92.2	95.7
Balanced 50–50	98.7	97.3	95.4	92.0	99.6
Unbalanced 80–20	99.3	89.3	97.4	46.8	66.5
MNIST, dense encoder					
Unbalanced 20–80	99.3	79.2	97.4	78.1	40.2
Balanced 50–50	98.8	53.4	95.4	71.7	32.1
Unbalanced 80–20	99.2	66.5	97.1	76.8	33.7
MNIST inverted target, spatial encoder					
Unbalanced 20–80	99.6	96.4	97.9	93.9	74.7
Balanced 50–50	96.7	96.6	91.7	91.7	74.7
Unbalanced 80–20	99.2	99.2	97.0	97.2	78.6
MNIST inverted target, dense encoder					
Unbalanced 20–80	99.4	66.9	97.5	33.6	34.9
Balanced 50–50	97.5	48.8	94.5	31.1	20.4
Unbalanced 80–20	97.7	61.9	91.4	54.2	14.5
MNIST flipped target, spatial encoder					
Unbalanced 20–80	99.6	80.2	98.1	29.9	52.5
Balanced 50–50	96.4	59.6	92.7	56.0	39.4
Unbalanced 80–20	98.0	75.3	94.3	66.4	36.8
MNIST flipped target, dense encoder					
Unbalanced 20–80	99.4	77.9	97.4	77.3	53.0
Balanced 50–50	91.2	51.0	80.2	53.4	18.3
Unbalanced 80–20	99.3	65.5	97.2	77.9	24.0

7.4.2. Cardiac CT/MRI dataset: MM-WHS

Our second demonstration uses CT/MRI scans from the Multi-Modal Whole Heart Segmentation dataset (MM-WHS, Zhuang and Shen, 2016). This heart segmentation dataset includes unpaired CT and MRI scans for 40 patients (20 CT and 20 MRI). Following Al Chanti and Mateus (2021), we used the ground-truth labels for four classes: left ventricle myocardium, left ventricle blood cavity, left atrium blood cavity, and ascending aorta.

We extracted 2D patches of 32 × 32 pixels, labeled with the class of the center pixel and balanced to have an equal number of samples per class. For our analysis, we converted this to a binary classification problem by grouping left atrium and left ventricle in one class, and ascending aorta and left ventricle myocardium in the other.

The experiments explored four domain adaptation scenarios: CT-to-MRI, MRI-to-CT, CT-to-inverted-CT, and MRI-to-inverted-MRI. In this dataset, the CT and MRI modalities provide complementary information, which makes the domain adaptation more challenging (some structures that are visible in CT are not visible in MRI, and vice versa). The experiments with inverted target domains avoid this complication (the source and target domains contain the same information), which makes it easier to observe the behavior of the domain adaptation algorithm.

Table 4
Results for experiments with BRATS data, showing mean validation performance averaged over 25 models.
Models were trained to convergence. Sparkline plots show the distribution of results for individual runs.

	Accuracy (%)		Confidence		Linear CKA
	Source	Target	Source	Target	Target
BRATS, spatial encoder, early join					
Balanced 50–50	73.9	62.7	65.6	57.0	61.4
BRATS, spatial encoder, late join					
Balanced 50–50	73.9	49.6	69.1	45.6	4.3
BRATS, dense encoder					
Balanced 50–50	77.4	51.3	66.9	45.3	2.4
BRATS, posterior join					
Balanced 50–50	77.5	50.5	69.3	46.2	0.5

Table 5
Results for experiments with MM-WHS data, showing mean validation performance averaged over 25 models.
Models were trained to convergence. Sparkline plots show the distribution of results for individual runs.

	Accuracy (%)		Confidence		Linear CKA
	Source	Target	Source	Target	Target
MM-WHS, spatial encoder, early join					
CT to MRI	61.1	55.7	45.8	23.9	74.9
MRI to CT	74.2	50.6	60.8	7.2	54.4
CT to inverted CT	60.2	60.1	43.6	41.5	74.8
MRI to inverted MRI	79.8	78.6	72.5	68.9	80.0
MM-WHS, spatial encoder					
CT to MRI	61.9	49.3	54.5	14.1	45.2
MRI to CT	76.7	50.4	62.4	3.4	29.9
CT to inverted CT	58.0	49.9	31.9	0.8	72.6
MRI to inverted MRI	79.5	47.4	73.1	37.5	39.6
MM-WHS, dense encoder					
CT to MRI	59.5	50.2	44.4	15.5	47.0
MRI to CT	67.2	50.4	35.5	2.1	18.3
CT to inverted CT	57.2	50.1	30.1	1.4	67.1
MRI to inverted MRI	78.4	50.2	66.6	26.5	41.8
MM-WHS, posterior join					
CT to MRI	57.0	49.9	27.0	3.5	19.8
MRI to CT	58.6	50.0	19.1	0.0	12.4
CT to inverted CT	56.1	50.0	23.0	0.0	59.1
MRI to inverted MRI	55.3	50.0	10.3	0.0	20.0

7.4.3. Architectures

We compared four models, all based on the same architecture but joining the source and target branches at different levels (Fig. 3, see the Appendix for full details: Figs. A.6 and A.7 for BRATS, Figs. A.8 and A.9 for MM-WHS). The *spatial encoder, early join* model joins the representations at an early spatial level (after the first pooling layer). This makes it relatively easy to join the domains if the domains are fairly similar, but also limits the complexity of the transformations that can be modeled. The *spatial encoder, late join* model joins the representations before the global pooling layer. This allows the model to learn more complex transformations, supporting larger differences between domains, but the increased complexity will also make it more difficult to learn the correct transformation. Because the representations are joined before global pooling, this architecture can still exploit spatial similarities. The *dense encoder* model joins the representations after global pooling, removing spatial information. The *posterior join* model joins the domain-specific branches only at the level of the final output. This model has the least information, and must link the domains based on the posterior distributions.

7.4.4. Results

Table 4 shows the results of these four models on the BRATS dataset. This task was more complicated than those in our previous experiments. The early-join spatial encoder achieved a reasonable target accuracy in a number of runs, but not all. The confidence and linear CKA scores suggest that this model also had modest success at identifying the clusters. The scores for the late-join spatial encoder and the non-spatial models were much worse. Neither the confidence, nor the accuracy on the target domain were very good, indicating that the domain adaptation failed to find clusters or link them between domains. The linear CKA scores are very low, indicating that the models learned very dissimilar representations.

Table 5 shows the results for the MM-WHS experiments. The adaptation between CT and MRI was challenging in both directions. The adaptation to an inverted CT or MRI was easier, but still far from perfect. Similar to the results for BRATS, the early-join spatial encoder obtained the highest domain adaptation scores: the target accuracy, target confidence, and linear CKA were higher for those models than for the others. This is especially clear for the inverted target domains, for which the early-join spatial encoder obtained target accuracies that were very close to the source accuracies. There is a substantial variability between results for different runs, illustrating the unpredictability of the domain adaptation outcome. There is also a clear difference between the scores on CT-to-MRI and MRI-to-CT, which suggests that the intensity differences going from CT to MRI were more favorable to the model than the reverse. On the other hand, the source accuracy on the MRI-to-CT experiment was higher, which indicates that the domain adaptation had a negative effect on the CT-to-MRI performance. Similar to BRATS, the late-join, dense, and posterior models performed worse.

Overall, the results for the BRATS and MM-WHS dataset suggest that spatial information was crucial for the models to learn a correct mapping between domains. In addition, the results for MM-WHS suggest that the domain adaptation process is sensitive to the intensity differences between CT and MRI.

Table A.6
Confusion matrices (%) for example runs on the synthetic ten dataset, with balanced (runs 1–3) or unbalanced (runs 4–6) data. Domain adversarial learning finds the correct class balance, but creates random combinations of clusters to do so.

Run	Prediction	Clusters from source domain										Clusters from target domain									
		1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
1	Class 1	9	10	9	10	9						9				9		10		10	10
	Class 2						10	9	10	10	9		9	10	9		9		10		
2	Class 1	10	9	10	9	10						10			9	10			9	9	
	Class 2						9	10	9	9	9		9	9			9	10			10
3	Class 1	10	10	9	9	9						9			10	10		9		9	
	Class 2						10	10	9	9	9		10	10			10		9		10
4	Class 1	10	9										10	10							
	Class 2			9	9	10	10	10	9	9	10	9			9	9	10	10	9	10	10
5	Class 1	9	9										10				10				
	Class 2			10	10	10	10	9	9	9	10	10		9	9	10		10	9	9	9
6	Class 1	9	9												9			9			
	Class 2			9	9	10	9	10	9	10	9	9	10	9		10	10		10	9	10

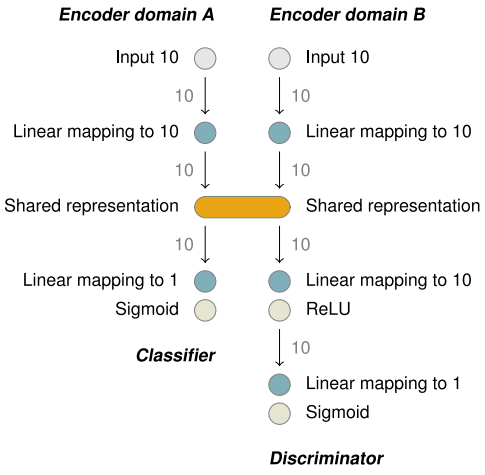


Fig. A.4. Network architecture for the synthetic experiments.

8. Discussion

In this paper we explored the limitations of unsupervised domain adaptation, using adversarial learning to learn domain-invariant representations. We addressed a common domain adaptation scenario where labeled training data is available for the source domain but not for the target domain, and where there are no paired samples that can be used to learn correspondences between domains. In this setting, adversarial domain adaptation attempts to learn a domain-invariant representation by aligning the source and target distributions in the latent feature space. We showed that this unsupervised distribution matching may lead to incorrect results, because there is no guarantee that similar samples in different domains will be mapped to similar latent representations. However, we also observed that domain-invariant properties of the data can introduce a bias that helps the model find the correct mapping. We identified four types of similarities that regularly occur in medical images.

8.1. Unsupervised domain adaptation without paired samples is flexible but unpredictable

In our experiments, we used models with domain-specific encoders. Using domain-specific encoders instead of a single, shared encoder allows the model to accommodate large differences between domains. This is convenient if the domains are very different, because the encoders can learn a domain-specific mapping for each domain. In comparison, a model with a single encoder is restricted to extracting domain-invariant features that have a similar appearance in both domains.

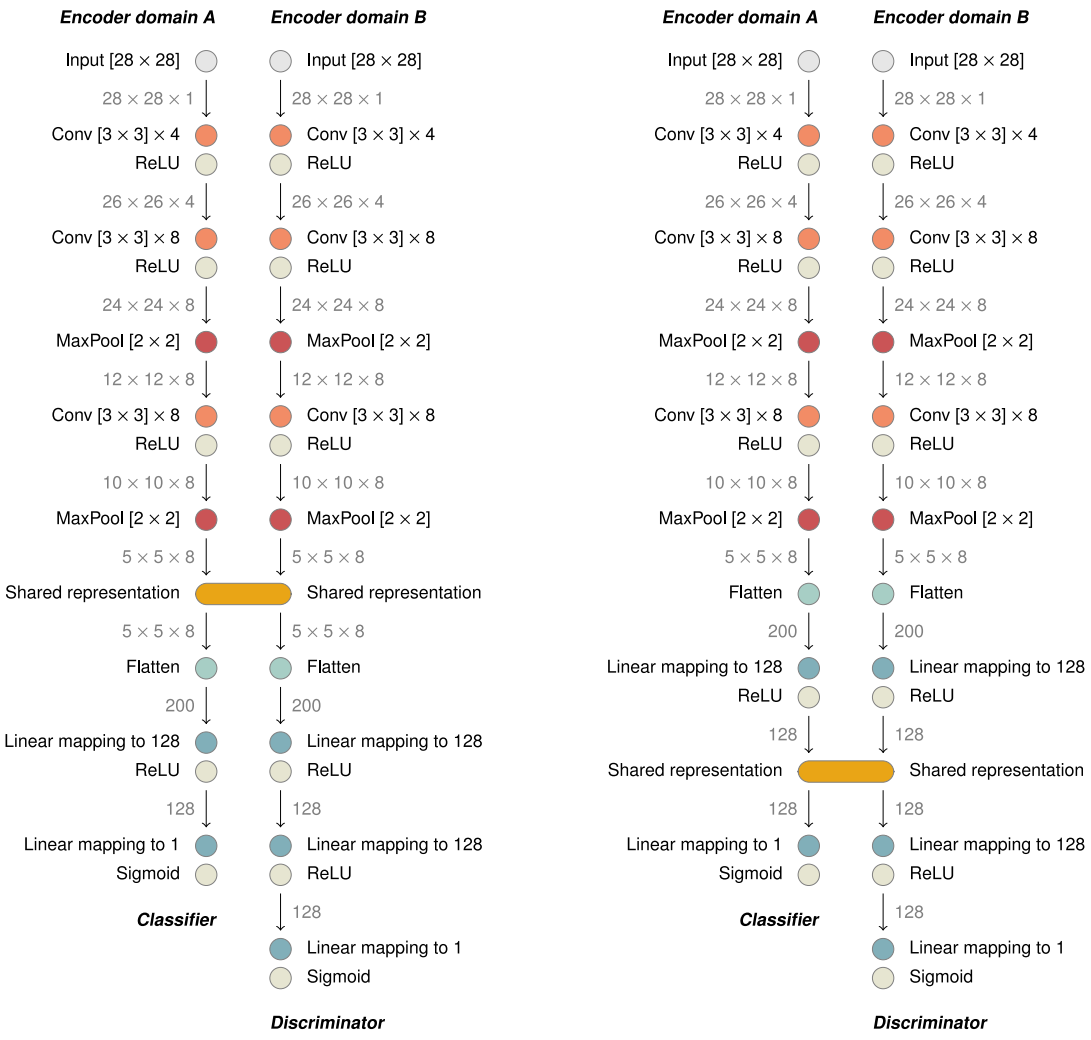
The flexibility afforded by the domain-specific encoders comes at a cost: without labeled target data or paired samples, it is difficult to link the domains correctly. In Section 4, we discussed that there are many possible ways to map samples between domains, and there is no guarantee that the model will automatically find the correct solution. The synthetic experiments in Section 7.2 showed a clear example of this problem: the models learned a random mapping that was either completely correct or completely wrong.

8.2. Similarities between domains may help or hinder the domain adaptation process

Despite the lack of guarantees, unsupervised domain adaptation can still succeed if the domains are sufficiently similar. In Section 5, we discussed four domain-invariant properties that are commonly seen in medical imaging data, and which may provide a useful source of domain adaptation bias:

- The model can use the class imbalance to identify classes, if this is similar between the source and target domains. This is more likely to work in datasets with fairly homogeneous classes, such as our synthetic example.
- The model can match classes based on average intensity, if this is similar in both domains. We saw evidence for an intensity-based bias in the experiments with synthetic and MNIST data, as well as on the CT/MRI scans in the MM-WHS dataset.
- The model can use the large-scale spatial similarities to match classes. This is sensitive to rotations and inversions, but can be very powerful if the images in both domains have a similar spatial structure. The convolutional feature extraction layers preserve the spatial arrangement of the input, if the encoding branches are joined at a spatial level. We observed that spatial information was important in our MNIST, BRATS, and MM-WHS experiments.
- The model might use local textures to match classes based on the strength of the textures in the image. This requires that the textures are comparable between domains, which might be difficult in more complex tasks, such as between CT and MRI. This effect is more difficult to measure, but we saw signs of this in the confidence scores of the dense encoders in some of our experiments.

Since many medical datasets exhibit some of these similarities, the domain adaptation process may be biased towards learning the correct mapping. Many domain adaptation approaches from the medical imaging literature rely on these similarities between domains explicitly, either by using an architecture with shared encoders or by introducing



(a) MNIST model: spatial encoder.

(b) MNIST model: dense encoder.

Fig. A.5. Network architectures for the MNIST experiments. The point of the division between domain-specific encoders and the shared classifier depends on the experiment.

additional constraints in the domain adaptation process. However, we found that these assumptions are also used implicitly in a model with domain-specific encoders.

8.3. Limitations and practical consequences

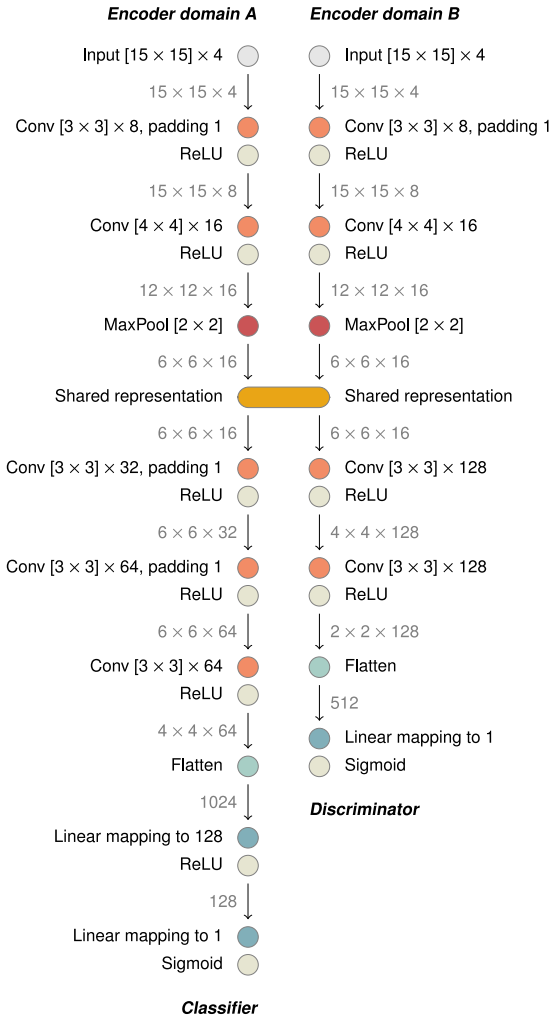
The internal behavior of domain adaptation methods is difficult to observe in practice. Our experiments on synthetic and MNIST data provided useful insights in the process, but the models and data were simpler than those in most real-world applications. The relatively homogeneous data allowed us to compute the subcluster-based metrics required for our analysis, but real data will be more heterogeneous and usually comes without subcluster labels. Our experiments on BRATS and MM-WHS used more realistic data, but were less transparent.

The observations in this paper were made on models using domain-specific encoders. While this allows a very flexible mapping between domains, it also makes it harder to learn a correct mapping. In contrast, models with shared encoders may be more likely to find a correct mapping if the domains are somewhat similar, but may have problems with larger differences between domains.

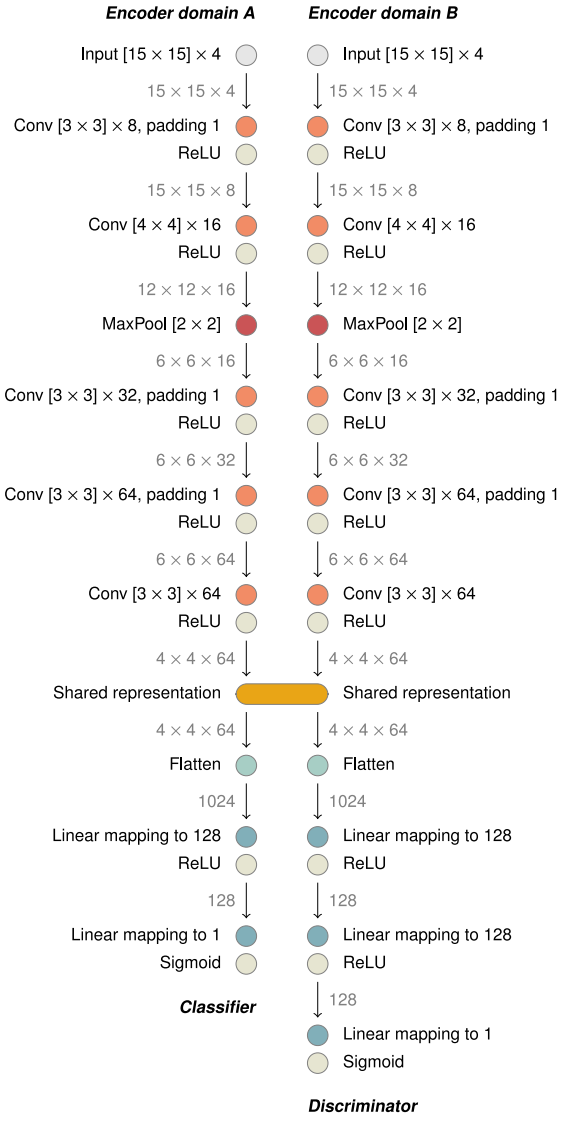
Our experiments on medical imaging datasets are based on patch-wise classification models. This resembles the pixel-wise classification of a segmentation task and is sufficient for our analysis, but is not competitive with the performance of more advanced state-of-the-art models. For example, it is likely that a specialized segmentation network such as a U-Net-like architecture (Ronneberger et al., 2015) could obtain a better segmentation result by improving the spatial consistency of the segmentation, but this would require the domains to be spatially similar.

In practice, there is often some form of spatial similarity between medical images from different sources, because they share similar anatomical structure. For applications where this can be ensured, a model that exploits these spatial similarities can often provide superior results. Image-to-image translation methods are a popular type of these methods, for example by using cycle consistency (Zhu et al., 2017), by including a segmentation objective (Wang and Zheng, 2022), or by using style transfer to separate content and appearance features (Yang et al., 2019; Chen et al., 2020b, 2021b).

Despite these limitations of our experiments, we believe that most of our conclusions also apply to more advanced models. Since there are no



(a) BRATS model: spatial encoder, early join.



(b) BRATS model: spatial encoder, late join.

Fig. A.6. Network architecture for the BRATS experiments with spatial encoders. The point of the division between domain-specific encoders and the shared classifier depends on the experiment.

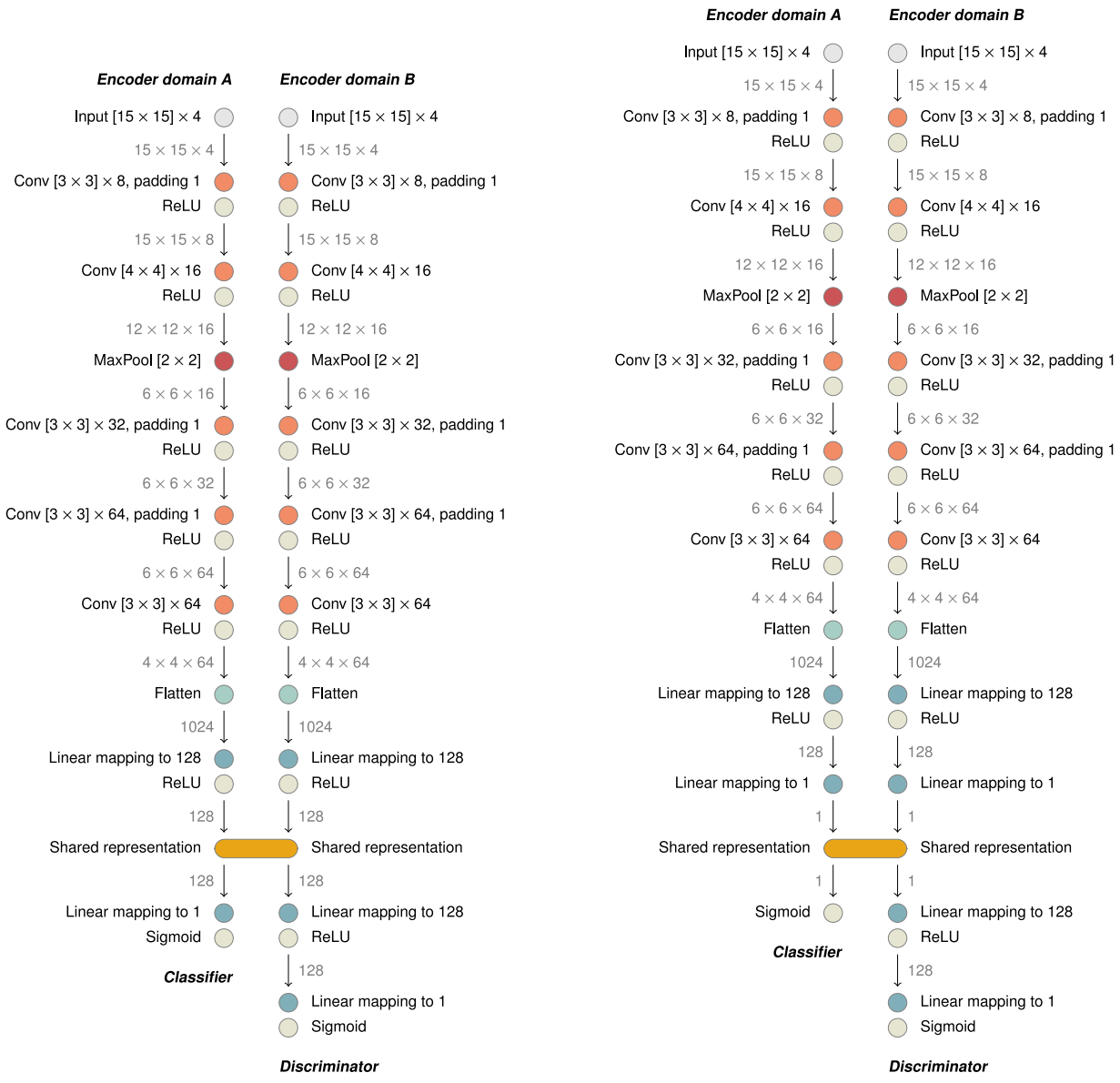
guarantees that unsupervised domain adaptation works in the general case, its success for specific applications must mean that the models exploit some underlying similarities in the data. The four assumptions discussed in Section 5 suggest what those similarities could be. We believe that many medical imaging tasks satisfy some or all of these assumptions, and suspect that this is why domain adaptation often succeeds.

It is important to be aware of these properties when applying domain adaptation to a new dataset. Even if the assumptions are not explicitly encoded in an auxiliary learning objective or constraint, they may still affect the outcome through implicit biases in the models. We would also like to note that this is not unique to domain adaptation at a feature level. Image-to-image translation methods such as CycleGAN, which constrain the translation to maintain the global spatial structure of the translated images, will face similar problems when translating local textures and intensities.

9. Conclusion

Learning unsupervised domain adaptation from unpaired samples is an ambitious goal, and to some extent it is surprising that it works at all. In this paper, we argued that successful unsupervised domain adaptation relies on similarities between domains. It is important to recognize these implicit assumptions, because they may influence the domain adaptation result. We explored several types of similarity that are common in medical images, and found that they can indeed help to push the domain adaptation in the right direction. Identifying potential implicit biases is a key step in obtaining reliable results.

However, even if those assumptions are satisfied, a correct domain adaptation is not guaranteed. In our experiments on the BRATS and MM-WHS datasets, unsupervised domain adaptation failed for anything but the simplest case. In practice, we suspect that unsupervised domain



(a) BRATS model: dense encoder.

(b) BRATS model: posterior join.

Fig. A.7. Network architecture for the BRATS experiments with dense encoders. The point of the division between domain-specific encoders and the shared classifier depends on the experiment.

adaptation can work well if domains are already similar, but needs additional constraints if they are not.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Gijs van Tulder and Marleen de Bruijne report financial support was provided by the Dutch Research Council (NWO). Marleen de Bruijne is a member of the Editorial Board of Medical Image Analysis.

Data availability

The experiments used publicly available datasets.

Acknowledgments

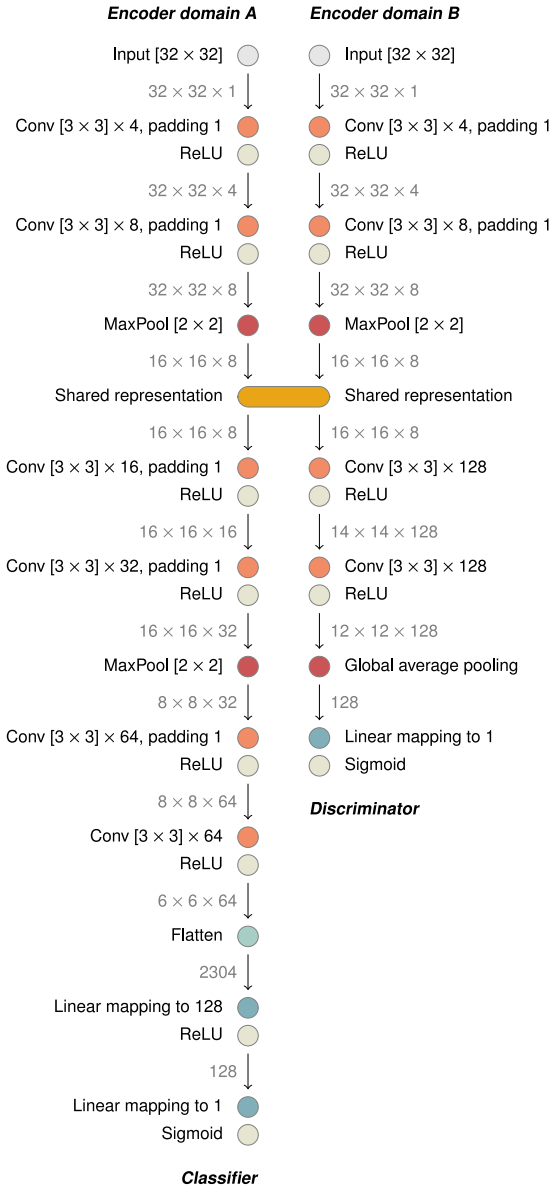
The authors received funding from the Dutch Research Council (NWO) with grant numbers 639.022.010 and VI.C.182.042.

Appendix. Implementation details

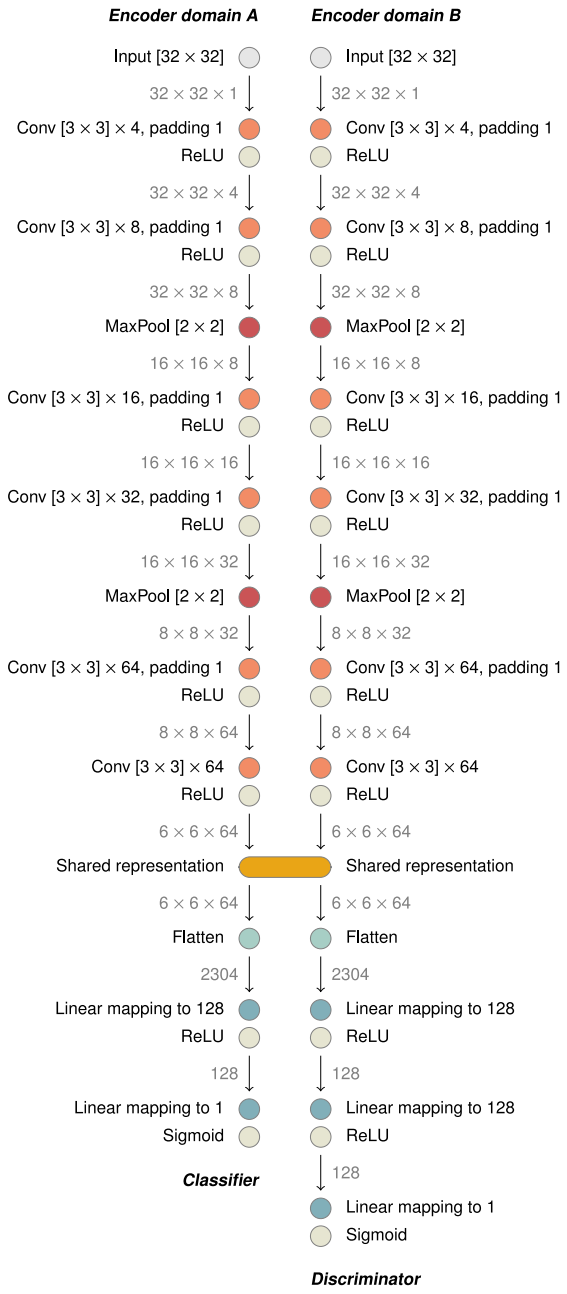
All experiments were implemented with PyTorch. In all experiments, domains A and B were trained with independent training samples from the same distribution. For the synthetic experiments, we used an infinite stream of random samples. For the MNIST experiments, we used the official training and test split.

For the BRATS experiment, we used the high-grade glioma subset and split the data in separate training, validation and testing sets, keeping samples from the same subject in a single subset. We used 80 subjects for training domain A, 80 subjects for training domain B, 30 subjects for validation, and 30 subjects for testing. For each subject, we selected patches centered on pixels from the ground-truth segmentation, while maintaining the class balance.

For the MM-WHS experiment, containing 40 patients with 20 patients for CT and 20 patients for MRI, we split the subjects for each modality in groups of 10/5/5 subjects for training, validation, and



(a) MM-WHS model: spatial encoder, early join.



(b) MM-WHS model: spatial encoder, late join.

Fig. A.8. Network architecture for the MM-WHS experiments with spatial encoders. The point of the division between domain-specific encoders and the shared classifier depends on the experiment.

testing. Similar to BRATS, we selected patches centered on pixels from the ground-truth segmentation, while maintaining the class balance.

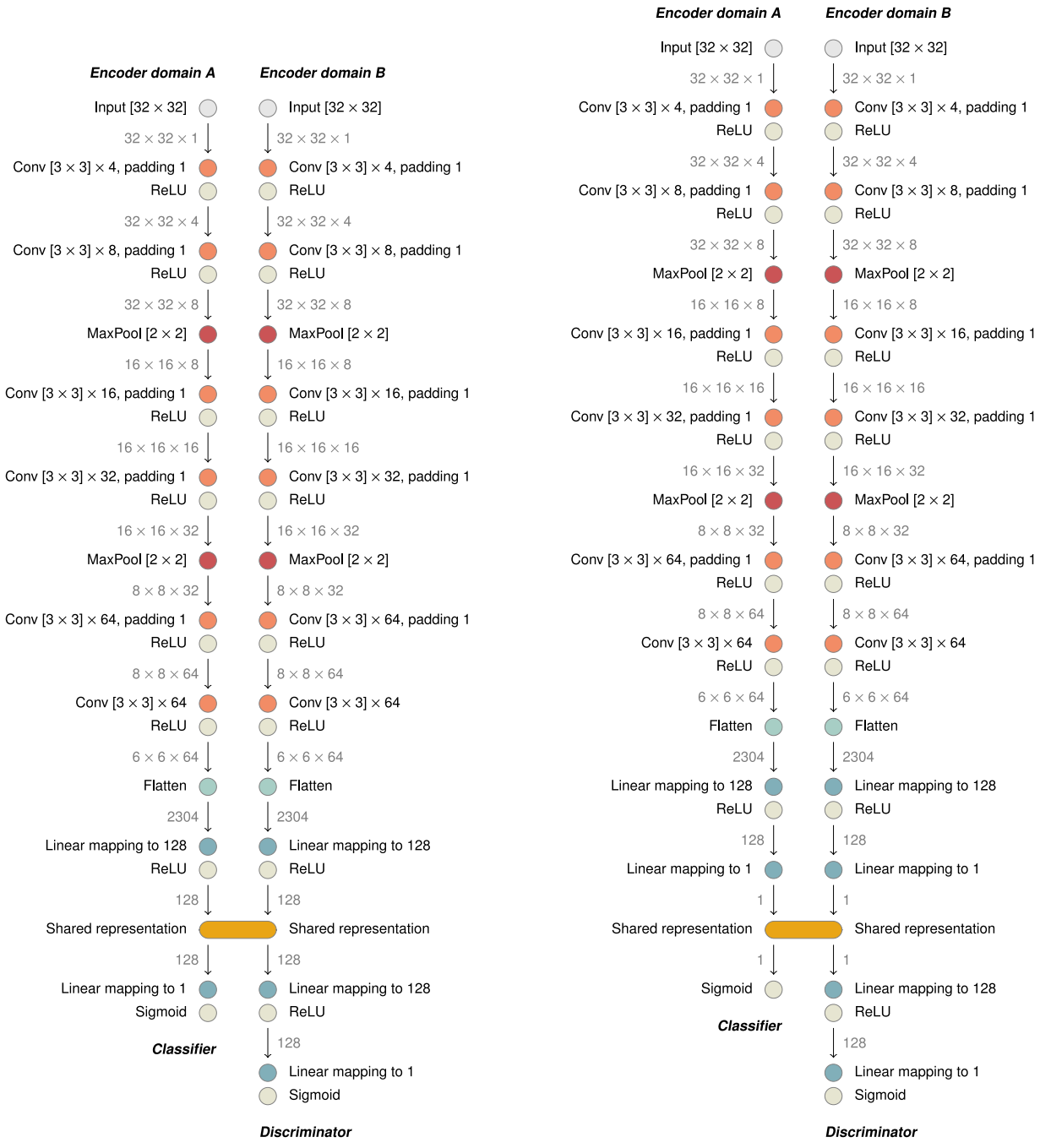
Our aim was to identify scenarios where domain adaptation could potentially work, but was unable to link the two domains. Consequently, we selected hyperparameters based on the results on domain A, while checking the confidence on domain B to ensure that the adaptation did not map all samples to a single class. Using the selected hyperparameters, we ran 25 experiments with different random initializations to obtain the results shown in the tables.

We fixed the weight of the classification term in the learning objective to $\lambda_{\text{class}} = 0.1$ for all experiments. For the discriminative, we chose one of $\lambda_{\text{disc}} \in \{0.3, 0.2, 0.1, 0.01, 0.001, 0.0001\}$ based on the performance on the source domain.

The learning rate was chosen from $\{0.001, 0.0005, 0.0001, 0.00001\}$. For the synthetic experiments, we used 0.001 for all experiments. For MNIST, BRATS, and MM-WHS, we used $0.001, 0.0005, 0.0001$ depending on the setting, but all three values gave similar results.

We optimized the models using Adam with a minibatch size of 128, for 200 epochs (MNIST), 150 epochs (MM-WHS), or 100 epochs (other experiments). This was sufficient for all networks to converge. We report the results at the end of the final epoch.

The source code for these experiments is available at https://vantulder.net/code/2023/uuda/.



(a) MM-WHS model: dense encoder.

(b) MM-WHS model: posterior join.

Fig. A.9. Network architecture for the MM-WHS experiments with dense encoders. The point of the division between domain-specific encoders and the shared classifier depends on the experiment.

References

Ackaouy, A., Courty, N., Vallée, E., Commowick, O., Barillot, C., Galassi, F., 2020. Unsupervised domain adaptation with optimal transport in multi-site segmentation of multiple sclerosis lesions from MRI data. Front. Comput. Neurosci. 14, http://dx.doi.org/10.3389/fncom.2020.00019.

Al Chanti, D., Mateus, D., 2021. OLVA: Optimal latent vector alignment for unsupervised domain adaptation in medical image segmentation. In: de Bruijne, M., Cattin, P.C., Cotin, S., Padoy, N., Speidel, S., Zheng, Y., Essert, C. (Eds.), Medical Image Computing and Computer Assisted Intervention – MICCAI 2021. Springer International Publishing, Cham, pp. 261–271. http://dx.doi.org/10.1007/978-3-030-87199-4_25, arXiv:2106.08188.

Bateson, M., Dolz, J., Kervadec, H., Lombaert, H., Ayed, I.B., 2021. Constrained domain adaptation for image segmentation. IEEE Trans. Med. Imaging 40, 1875–1887. http://dx.doi.org/10.1109/TMI.2021.3067688.

Bateson, M., Kervadec, H., Dolz, J., Lombaert, H., Ben Ayed, I., 2020. Source-Relaxed Domain Adaptation for image segmentation. In: Martel, A.L., Abolmaesumi, P., Stoyanov, D., Mateus, D., Zuluaga, M.A., Zhou, S.K., Racoceanu, D., Joskowicz, L. (Eds.), Medical Image Computing and Computer Assisted Intervention – MICCAI 2020. Springer International Publishing, Cham, pp. 490–499. http://dx.doi.org/10.1007/978-3-030-59710-8_48.

Ben-David S. Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., Vaughan, J.W., 2010a. A theory of learning from different domains. Mach. Learn. 79, 151–175. http://dx.doi.org/10.1007/s10994-009-5152-4.

Ben-David S. Luu, T., Lu, T., Pál, D., 2010b. Impossibility theorems for domain adaptation. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings. pp. 129–136.

Bian, C., Yuan, C., Wang, J., Li, M., Yang, X., Yu, S., Ma, K., Yuan, J., Zheng, Y., 2020. Uncertainty-aware domain alignment for anatomical structure segmentation. Med. Image Anal. 64, 101732. http://dx.doi.org/10.1016/j.media.2020.101732.

Cai, J., Zhang, Z., Cui, L., Zheng, Y., Yang, L., 2019. Towards cross-modal organ translation and segmentation: A cycle- and shape-consistent generative adversarial network. Med. Image Anal. 52, 174–184. http://dx.doi.org/10.1016/j.media.2018.12.002.

Chen, C., Dou, Q., Chen, H., Qin, J., Heng, P.A., 2019. Synergistic image and feature adaptation: Towards cross-modality domain adaptation for medical image segmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 33, pp. 865–872. http://dx.doi.org/10.1609/aaai.v33i01.3301865, arXiv:1901.08211.

Chen, C., Dou, Q., Chen, H., Qin, J., Heng, P.A., 2020a. Unsupervised bidirectional cross-modality adaptation via deeply synergistic image and feature alignment for medical image segmentation. IEEE Trans. Med. Imaging 39, 2494–2505. http://dx.doi.org/10.1109/TMI.2020.2972701.

Chen, X., Lian, C., Wang, L., Deng, H., Kuang, T., Fung, S.H., Gateno, J., Shen, D., Xia, J.J., Yap, P.T., 2021b. Diverse data augmentation for learning image segmentation with cross-modality annotations. Med. Image Anal. 71, 102060. http://dx.doi.org/10.1016/j.media.2021.102060.

Chen, X., Lian, C., Wang, L., Deng, H., Kuang, T., Fung, S., Gateno, J., Yap, P.T., Xia, J.J., Shen, D., 2021a. Anatomy-regularized representation learning for cross-modality medical image segmentation. IEEE Trans. Med. Imaging 40, 274–285. http://dx.doi.org/10.1109/TMI.2020.3025133.

Chen, C., Ouyang, C., Tarroni, G., Schlemper, J., Qiu, H., Bai, W., Rueckert, D., 2020b. Unsupervised multi-modal style transfer for cardiac MR segmentation. In: Pop, M., Sermesant, M., Camara, O., Zhuang, X., Li, S., Young, A., Mansi, T., Suinesiaputra, A. (Eds.), Statistical Atlases and Computational Models of the Heart. Multi-Sequence CMR Segmentation, CRT-EPiggy and LV Full Quantification Challenges. Springer International Publishing, Cham, pp. 209–219. http://dx.doi.org/10.1007/978-3-030-39074-7_22.

Cohen, J.P., Luck, M., Honari, S., 2018. Distribution matching losses can hallucinate features in medical image translation. In: Frangi, A.F., Schnabel, J.A., Davatzikos, C., Alberola-López, C., Fichtinger, G. (Eds.), Medical Image Computing and Computer Assisted Intervention – MICCAI 2018. Springer International Publishing, Cham, pp. 529–536. http://dx.doi.org/10.1007/978-3-030-00928-1_60, arXiv:1805.08841.

Cui, Z., Li, C., Du, Z., Chen, N., Wei, G., Chen, R., Yang, L., Shen, D., Wang, W., 2021. Structure-driven unsupervised domain adaptation for cross-modality cardiac segmentation. IEEE Trans. Med. Imaging 40, 3604–3616. http://dx.doi.org/10.1109/TMI.2021.3090432.

Dar, S.U., Yurt, M., Karacan, L., Erdem, A., Erdem, E., Çukur, T., 2019. Image synthesis in multi-contrast MRI with conditional generative adversarial networks. IEEE Trans. Med. Imaging 38, 2375–2388. http://dx.doi.org/10.1109/TMI.2019.2901750.

de Bel, T., Bokhorst, J.M., van der Laak, J., Litjens, G., 2021. Residual cyclegan for robust domain transformation of histopathological tissue slides. Med. Image Anal. 70, 102004. http://dx.doi.org/10.1016/j.media.2021.102004.

Ganin, Y., Lempitsky, V., 2015. Unsupervised domain adaptation by backpropagation. In: Proceedings of the 32nd International Conference on Machine Learning. ICML, PMLR, pp. 1180–1189, arXiv:1409.7495.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., Lempitsky, V., 2017. Domain-Adversarial Training of neural networks. In: Csurka, G. (Ed.), Domain Adaptation in Computer Vision Applications. Springer International Publishing, Cham, pp. 189–209. http://dx.doi.org/10.1007/978-3-319-58347-1_10.

Gao, Y., Liu, Y., Wang, Y., Shi, Z., Yu, J., 2019. A universal intensity standardization method based on a many-to-one weak-paired cycle generative adversarial network for magnetic resonance images. IEEE Trans. Med. Imaging 38, 2059–2069. http://dx.doi.org/10.1109/TMI.2019.2894692.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets. In: Advances in Neural Information Processing Systems. NIPS, Curran Associates, Inc..

Gretton, A., Smola, A., Huang, J., Schmittfull, M., Borgwardt, K., Schölkopf, B., 2008. Covariate shift by kernel mean matching. In: Dataset Shift in Machine Learning. The MIT Press, http://dx.doi.org/10.7551/mitpress/9780262170055.003.0008.

Guan, H., Liu, M., 2021. Domain Adaptation for Medical Image Analysis: A Survey. [cs, eess] arXiv:2102.09508.

Guan, H., Liu, Y., Yang, E., Yap, P.T., Shen, D., Liu, M., 2021. Multi-site MRI harmonization via attention-guided deep domain adaptation for brain disorder identification. Med. Image Anal. 71, 102076. http://dx.doi.org/10.1016/j.media.2021.102076.

Hoffman, J., Tzeng, E., Park, T., Zhu, J.Y., Isola, P., Saenko, K., Efros, A., Darrell, T., 2018. CyCADA: cycle-consistent adversarial domain adaptation. In: Proceedings of the 35th International Conference on Machine Learning. ICML, PMLR, pp. 1989–1998.

Hu, D., Zhang, H., Wu, Z., Wang, F., Wang, L., Smith, J.K., Lin, W., Li, G., Shen, D., 2020. Disentangled-Multimodal Adversarial Autoencoder: Application to infant age prediction with incomplete multimodal neuroimages. IEEE Trans. Med. Imaging 39, 4137–4149. http://dx.doi.org/10.1109/TMI.2020.3013825.

Jiao, J., Namburete, A.I.L., Papageorghiou, A.T., Noble, J.A., 2020. Self-Supervised Ultrasound to MRI fetal brain image synthesis. IEEE Trans. Med. Imaging 39, 4413–4424. http://dx.doi.org/10.1109/TMI.2020.3018560.

Ju, L., Wang, X., Zhao, X., Bonnington, P., Drummond, T., Ge, Z., 2021. Leveraging regular fundus images for training uwf fundus diagnosis models via adversarial learning and pseudo-labeling. IEEE Trans. Med. Imaging 40, 2911–2925. http://dx.doi.org/10.1109/TMI.2021.3056395.

Kamnitsas, K., Baumgartner, C., Ledig, C., Newcombe, V., Simpson, J., Kane, A., Menon, D., Nori, A., Criminisi, A., Rueckert, D., Glocker, B., 2017. Unsupervised domain adaptation in brain lesion segmentation with adversarial networks. In: Niethammer, M., Styner, M., Aylward, S., Zhu, H., Oguz, I., Yap, P.T., Shen, D. (Eds.), Information Processing in Medical Imaging (IPMI). Springer International Publishing, Cham, pp. 597–609. http://dx.doi.org/10.1007/978-3-319-59050-9_47, arXiv:1612.08894.

Koohbanani, N.A., Unnikrishnan, B., Khurram, S.A., Krishnaswamy, P., Rajpoot, N., 2021. Self-Path: Self-supervision for classification of pathology images with limited annotations. IEEE Trans. Med. Imaging 40, 2845–2856. http://dx.doi.org/10.1109/TMI.2021.3056023.

Kornblith, S., Norouzi, M., Lee, H., Hinton, G., 2019. Similarity of neural network representations revisited. In: Proceedings of the 36th International Conference on Machine Learning (ICML). PMLR, pp. 3519–3529, arXiv:1905.00414.

Li, H., Han, H., Li, Z., Wang, L., Wu, Z., Lu, J., Zhou, S.K., 2020a. High-resolution chest X-ray bone suppression using unpaired CT structural priors. IEEE Trans. Med. Imaging 39, 3053–3063. http://dx.doi.org/10.1109/TMI.2020.2986242.

Li, H., Loehr, T., Wiestler, B., Zhang, J., Menze, B.H., 2020b.. E-UDA: Efficient unsupervised domain adaptation for cross-site medical image segmentation. arXiv:2001.09313v1.

Liang, J., Hu, D., Feng, J., 2020. Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation. In: Proceedings of the 37th International Conference on Machine Learning. PMLR, pp. 6028–6039, arXiv:2001.09313.

Liu, X., Guo, X., Liu, Y., Yuan, Y., 2021. Consolidated domain adaptive detection and localization framework for cross-device colonoscopic images. Med. Image Anal. 71, 102052. http://dx.doi.org/10.1016/j.media.2021.102052.

Luo, L., Yu, L., Chen, H., Liu, Q., Wang, X., Xu, J., Heng, P.A., 2020. Deep mining external imperfect data for chest X-ray disease screening. IEEE Trans. Med. Imaging 39, 3583–3594. http://dx.doi.org/10.1109/TMI.2020.3000949.

Menze, B.H., Jakab, A., Bauer, S., Kalpathy-cramer, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., Lanczi, L., Gerstner, E., Weber, M.a., Arbel, T., Avants, B.B., Ayache, N., Buendia, P., Collins, D.L., Cordier, N., Corso, J.J., Criminisi, A., Das, T., Durst, C.R., Dojat, M., Doyle, S., Festa, J., Forbes, F., Geremia, E., Glocker, B., Golland, P., Guo, X., Hamamci, A., Iftekharuddin, K.M., Jena, R., John, N.M., Meier, R., Precup, D., Price, S.J., Riklin-raviv, S.M.S., Ryan, M., Schwartz, L., Shin, H.c., Shotton, J., Silva, C.a., Sousa, N., Subbanna, N.K., Szekely, G., Taylor, T.J., Thomas, O.M., Tustison, N.J., Unal, G., Vasseur, F., Wintermark, M., Ye, D.H., Zhao, L., Zhao, B., Zikic, D., Prastawa, M., Reyes, M., Leemput, K.V., 2015. The multimodal brain tumor image segmentation benchmark (BRATS). IEEE Trans. Med. Imaging 34, 1993–2024. http://dx.doi.org/10.1109/TMI.2014.2377694.

Pei, C., Wu, F., Huang, L., Zhuang, X., 2021. Disentangle domain features for cross-modality cardiac image segmentation. Med. Image Anal. 71, 102078. http://dx.doi.org/10.1016/j.media.2021.102078.

Ren, M., Dey, N., Fishbaugh, J., Gerig, G., 2021. Segmentation-Renormalized Deep Feature Modulation for unpaired image harmonization. IEEE Trans. Med. Imaging 40, 1519–1530. http://dx.doi.org/10.1109/TMI.2021.3059726.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: Convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Springer International Publishing, Cham, pp. 234–241. http://dx.doi.org/10.1007/978-3-319-24574-4_28.

Shen, Y., Sheng, B., Fang, R., Li, H., Dai, L., Stolte, S., Qin, J., Jia, W., Shen, D., 2020. Domain-invariant interpretable fundus image quality assessment. Med. Image Anal. 61, 101654. http://dx.doi.org/10.1016/j.media.2020.101654.

Tomar, D., Lortkipanidze, M., Vray, G., Bozorgtabar, B., Thiran, J.P., 2021. Self-Attentive Spatial Adaptive Normalization for cross-modality domain adaptation. IEEE Trans. Med. Imaging 40, 2926–2938. http://dx.doi.org/10.1109/TMI.2021.3059265.

Tomczak, A., Ilic, S., Marquardt, G., Engel, T., Forster, F., Navab, N., Albarqouni, S., 2021. Multi-Task Multi-Domain Learning for digital staining and classification of leukocytes. IEEE Trans. Med. Imaging 40, 2897–2910. http://dx.doi.org/10.1109/TMI.2020.3046334.

Tzeng, E., Hoffman, J., Saenko, K., Darrell, T., 2017. Adversarial discriminative domain adaptation. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2962–2971. http://dx.doi.org/10.1109/CVPR.2017.316, arXiv:1702.05464.

Wang, S., Yu, L., Yang, X., Fu, C.W., Heng, P.A., 2019. Patch-Based Output Space Adversarial Learning for joint optic disc and cup segmentation. IEEE Trans. Med. Imaging 38, 2485–2495. http://dx.doi.org/10.1109/TMI.2019.2899910.

Wang, R., Zheng, G., 2022. Cycmis: cycle-consistent cross-domain medical image segmentation via diverse image augmentation. Med. Image Anal. 76, 102328. http://dx.doi.org/10.1016/j.media.2021.102328.

Wolterink, J.M., Seevinck, P.R., Dinkla, A.M., 2017. MR-to-CT synthesis using cycle-consistent generative adversarial networks. In: Med-NIPS.

Wu, F., Zhuang, X., 2021. Unsupervised domain adaptation with variational approximation for cardiac segmentation. IEEE Trans. Med. Imaging 40, 3555–3567. http://dx.doi.org/10.1109/TMI.2021.3090412.

Yang, J., Dvornek, N.C., Zhang, F., Chapiro, J., Lin, M., Duncan, J.S., 2019. Unsupervised domain adaptation via disentangled representations: Application to cross-modality liver segmentation. In: Shen, D., Liu, T., Peters, T.M., Staib, L.H., Essert, C., Zhou, S., Yap, P.T., Khan, A. (Eds.), Medical Image Computing and Computer Assisted Intervention – MICCAI 2019. Springer International Publishing, Cham, pp. 255–263. http://dx.doi.org/10.1007/978-3-030-32245-8_29.

Yang, H., Sun, J., Carass, A., Zhao, C., Lee, J., Xu, Z., Prince, J., 2018. Unpaired brain MR-to-CT synthesis using a structure-constrained cyclegan. In: Stoyanov, D., Taylor, Z., Carneiro, G., Syeda-Mahmood, T., Martel, A., Maier-Hein, L., Tavares, J.M.R., Bradley, A., Papa, J.P., Belagiannis, V., Nascimento, J.C., Lu, Z., Conjeti, S., Moradi, M., Greenspan, H., Madabhushi, A. (Eds.), Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support. Springer International Publishing, Cham, pp. 174–182. http://dx.doi.org/10.1007/978-3-030-00889-5_20.

Yu, B., Zhou, L., Wang, L., Shi, Y., Fripp, J., Bourgeat, P., 2020. Sample-Adaptive GANs: Linking global and local mappings for cross-modality MR image synthesis. IEEE Trans. Med. Imaging 39, 2339–2350. http://dx.doi.org/10.1109/TMI.2020.2969630.

Zhao, H., Combes, R.T.D., Zhang, K., Gordon, G., 2019. On learning invariant representations for domain adaptation. In: Proceedings of the 36th International Conference on Machine Learning (ICML). PMLR, pp. 7523–7532.

Zhou, B., Augenfeld, Z., Chapiro, J., Zhou, S.K., Liu, C., Duncan, J.S., 2021. Anatomy-guided multimodal registration by learning segmentation without ground truth: Application to intraprocedural CBCT/MR liver segmentation and registration. Med. Image Anal. 71, 102041. http://dx.doi.org/10.1016/j.media.2021.102041.

Zhu, J.Y., Park, T., Isola, P., Efros, A.A., 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 2242–2251. http://dx.doi.org/10.1109/ICCV.2017.244, arXiv:1703.10593.

Zhuang, X., Shen, J., 2016. Multi-scale patch and multi-modality atlases for whole heart segmentation of MRI. Med. Image Anal. 31, 77–87. http://dx.doi.org/10.1016/j.media.2016.02.006.

Zhuang, X., Xu, J., Luo, X., Chen, C., Ouyang, C., Rueckert, D., Campello, V.M., Lekadir, K., Vesal, S., RaviKumar, N., Liu, Y., Luo, G., Chen, J., Li, H., Ly, B., Sermesant, M., Roth, H., Zhu, W., Wang, J., Ding, X., Wang, X., Yang, S., Li, L., 2022. Cardiac segmentation on late gadolinium enhancement MRI: A benchmark study from multi-sequence cardiac MR segmentation challenge. Med. Image Anal. 81, 102528. http://dx.doi.org/10.1016/j.media.2022.102528.