Bidirectional Adversarial Domain Adaptation with Semantic Consistency

Yaping Zhang[1,2] *, Shuai Nie[1], Shan Liang[1], and Wenju Liu[1]**

[1] Institute of Automation, Chinese Academy of Sciences, China
[2] University of Chinese Academy of Sciences, China
{yaping.zhang, shuai.nie, sliang, lwj}@nlpr.ia.ac.cn

Abstract. Unsupervised domain adaptation (DA) aims to utilize the well-annotated source domain data to recognize the unlabeled target domain data that usually have a large domain shift. Most existing DA methods are developed to align the high-level feature-space distribution between the source and target domains, while neglecting the semantic consistency and low-level pixel-space information. In this paper, we propose a novel bidirectional adversarial domain adaptation (BADA) method to simultaneously adapt the pixel-level and feature-level shifts with semantic consistency. To keep semantic consistency, we propose a soft label-based semantic consistency constraint, which takes advantage of the well-trained source classifier during bidirectional adversarial mappings. Furthermore, the semantic consistency has been first analyzed during the domain adaptation with regard to both qualitative and quantitative evaluation. Systematic experiments on four benchmark datasets show that the proposed BADA achieves the state-of-the-art performance.

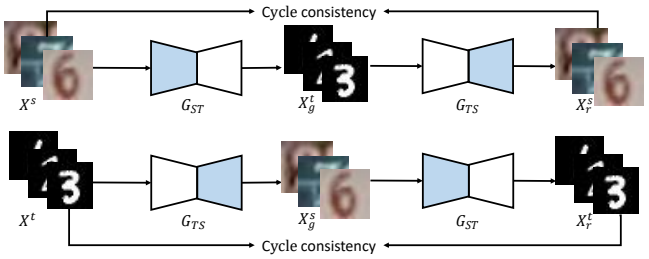
Keywords: Domain adaptation · GAN · unsupervised learning.

1 Introduction

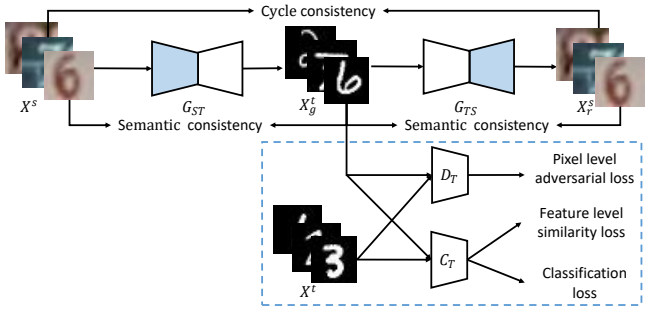
Deep learning has shown great success in multimedia analysis by learning discriminative representations from massive labeled data [9,7]. However, collecting the well-annotated datasets is exceedingly expensive. A promising alternative is to take full advantage of labeled data from an easily available source domain. Unfortunately, the inevitable domain shifts between the source and target domain limit the generalization of models. To alleviate this issue, recent domain adaptation methods try to align the feature distribution [4,29], which focus on minimizing the distance between the source and target feature domain.However, the feature-level alignment methods suffer two limitations: (1) feature-level alignment is hard to sufficiently transfer knowledge from the source domain to the target domain, due to missing the low-level pixel-space variance, which is critical to the generalization of the model; (2) the measure of feature-level difference fails to consider the semantic consistency during the alignment, and it is difficult to directly observe whether the transferred knowledge is reasonable.

Adversarial pixel-level domain adaptation [21] has shown great potential recently, which tries to align the raw pixel-level distribution between two domains. Specifically, pixel-level domain adaptation tries to map images from the source domain to appear as if they were sampled from the target domain, while keeping their original contents. The existing adversarial pixel-level domain

* The first author is a student.
** Corresponding author.



(a) Structure of CycleGAN



(b) Structure of the proposed BADA

Fig. 1: (a) Structure of CycleGAN:Cycle consistency only ensures the reconstruction of original content, where the middle mapping suffers label flipping. For example, the source image $\mathbf{X}^s$ is with label "6", while the transferred target image inconsistently belongs to label "3", hence it cannot be used to train a new target classifier. (b) Structure of the proposed BADA method: a generator G_{ST} that maps source domain images $\mathbf{X}^s$ to adapted target image $\mathbf{X}_g^t$, and an another inverted generator G_{TS} that generates the reconstructed image $\mathbf{X}_r^s$ as if from original source domain, while keeps cycle consistency and semantic consistency. For example, the transferred target image keeps the label "6", and can be used for training a new target classifier C_T. The target discriminator D_T is to distinguish the generated target images $\mathbf{X}_g^t$ from unpaired real target image X^t, which offers the guidance for generators.

adaptation is achieved by learning a unidirectional pixel-level mapping with unpaired images, which must maintain similar foregrounds between two domains to provide training stability.

Cycle-consistent adversarial network (CycleGAN) [28] introduces a pair of bidirectional mappings with cycle consistency to relax the strong assumption that two domains must have similar contents to capture larger domain shifts. The cycle consistency loss ensures that an image translated from one domain to another domain can be reconstructed to original domain. It shows compelling results on unpaired image-to-image translation tasks. However, CycleGAN *cannot* guarantee that the semantic contents are preserved during the translating process. As shown in Fig. 1 (a), CycleGAN suffers from random label flipping, that is, *lack of semantic consistency*.

To overcome the shortcoming of CycleGAN in the domain adaptation task, we proposed a novel Bidirectional Adversarial Domain Adaptation (BADA) model. As shown in Fig. 1 (b), BADA contains a pair of bidirectional reversible mappings: one generator G_{ST} maps source domain images $\mathbf{X}^s$ to the adapted target images $\mathbf{X}_g^t$, and another inverted generator G_{TS} that reconstructs adapted images back to the source domain, while keep cycle consistency and semantic consistency. The adapted target images $\mathbf{X}_g^t$ not only possess the style of the target domain, but also inherit the labels from the source domain. And thus the adapted target images $\mathbf{X}_g^t$ can be used to learn a supervised target classifier C_T. Furthermore, through the coordination between

the pixel-level adversarial loss and the feature-level similarity loss, the target classifier C_T is able to capture both the low-level and high-level shifts between the source domain and target domain. What's more, BADA is under the guidance of a soft label-based semantic consistency constraint, which takes advantage of semantic information during bidirectional mappings and is superior to unidirectional semantic consistency in CyCADA [8] and SBADA-GAN [20]. We summarize our contributions as follows:

- We propose a novel BADA method to jointly consider the pixel-level and feature-level domain adaptation with semantic consistency. The pixel-level adaptation preserves more detail information and is easily visualized, while the feature-level adaptation could capture more high-level domain-invariant representations.
- We propose a soft label-based semantic consistency constraint considering semantic information during bidirectional mappings, which effectively solves the random label flipping problem that is suffered by CycleGAN, and we analyze the semantic consistency with regard to both qualitative and quantitative evaluation for the first time.
- The proposed BADA significantly outperforms the state-of-the-art domain adaptation methods on some benchmark datasets, which shows that the proposed semantic consistency constraint, as well as the joint consideration of the pixel-level and feature-level domain adaptation can improve the domain adaptation ability.

2 Related Work

Existing methods generally aim to reduce domain shifts by minimizing the distance of feature distribution [4,29,26] between the source domain and target domain. The measure of distance can be roughly divided into maximum mean discrepancy (MMD) [14,2], correlation distances [22,23], deep reconstruction loss [6] or an adversarial loss [5,13,25,26]. While there are so many feature-level domain adaption methods, we mainly focus on the MMD-based and adversarial-loss based methods, which are highly related to our work. Maximum Mean Discrepancy (MMD) based methods [14,2] are to learn domain-invariant features by computing the norm of the difference between two domain means. The Deep Adaptation Network (DAN) [14] applies MMD to the feature layers of deep neural networks, effectively inducing a high-level feature alignment. Other methods chose an adversarial loss to measure the domain shifts between the learned features [25,26,3], which introduce an extra domain discriminator to encourage features not being distinguished between two domains. Adversarial loss based methods could be further divided into discriminative methods and generative methods. The adversarial discriminative methods [5,25] consider the feature alignment only, while adversarial generative domain adaptation methods [13,24] try to utilize a weight sharing constraint to learn a joint multi-domains distribution with the reconstruction of target domain. However, the performance of feature-level domain adaptation method is far from purely supervised methods, due to the lack of ability to capture pixel-level domain shifts. Recently, pixel-level domain adaptation methods have shown the huge potential [1,17,8]. Unsupervised Pixel-level Domain Adaptation (PixelDA) [1] adapts the source-domain images to appear as if drawn from the target domain, and achieve surprising results on some unsupervised domain adaptation task. While pixelDA has a strong assumption that the source domain and target domain must share many similar foregrounds limiting larger domain shifts.

In contrast, cycle-consistency loss based network [28,11] shows amazing results on unpaired image-to-image translation by a pair of dual pixel-level mappings, which do not need similar foregrounds and instead simply ensure that the translated images could be reconstructed back to their original domains with identical contents. However, they fails to keep the semantic consistency during the conversion process. Motivated by this, the proposed BADA model considers the unpaired pixel-level translation with a novel semantic consistency constraint for unsupervised domain adaptation. We note that the motivation of CyCADA [8] and SBADA-GAN [20]

are similar to ours. However, we solve the label flipping problem from different perspective. Compared to CyCADA and SBADA-GAN, we propose a more effective semantic consistency constraint, where we focus on the bidirectional reversible semantic consistency during the unpaired pixel-level mappings. Furthermore, we combine a simple but effective MMD feature-level domain adaptation method to boost performance. While CyCADA needs an extra discriminator neural network and SBADA-GAN needs to combine the source and target classifier for the final prediction. Moreover, we firstly analyze the semantic consistency, with regard to both qualitative and quantitative evaluation, during the domain adaptation.

3 The Proposed Model

3.1 Formulations

Suppose that there are N^s annotated source-domain samples $\mathbf{X}^s = \{{\mathbf{x}_s}^i\}_{i=0}^{N^s}$ with labels $\mathbf{Y}^s = \{{\mathbf{y}_s}^i\}_{i=0}^{N^s}$ and N^t unlabeled target-domain samples $\mathbf{X}^t = \{{\mathbf{x}_t}^i\}_{i=0}^{N^t}$. With the well-annotated source data, we could learn an optimized source classifier C_S parameterized θ_{C_S} by minimizing a standard supervised classification loss expressed as:

$$L_{cls}(C_S; \mathbf{X}^s, \mathbf{Y}^s) = E_{(\mathbf{x}_s, \mathbf{y}_s)\sim(\mathbf{X}^s, \mathbf{Y}^s)} \left[-{\mathbf{y}_s}^\top \log(\sigma(C_S(\mathbf{x}_s; \theta_{C_S}))) \right], \tag{1}$$

where $\mathbf{y}_s$ is the one-hot vector of the class label, and $\sigma(\cdot)$ denotes the softmax function.

However, the trained source classifier C_S is hard to perform well on the target domain, due to the inevitable shifts across the different domains. Our model is to adapt images from the source domain to appear as if they were drawn from the target domain by learning a discriminative mapping, and then we could use the generated labeled target domain images to train a new target classifier C_T as if the training and test data were from the same distribution. Unfortunately, lack of the paired images, the key semantic content is hard to keep consistent by the unidirectional pixel-to-pixel mapping from the source domain to the target domain. To alleviate this issue, we introduce two reversible mappings: a generator G_{ST} that maps a source domain image $\mathbf{x}_s$ to an adapted target image $\mathbf{x}_t^g = G_{ST}(\mathbf{x}_s)$, and an another inverted generator G_{TS} that makes a target domain image back to the source domain, ending up the same semantic content.

To ensure that learnt pixel-level mappings are semantic consistent between the source and target domain, we introduce four different losses: a *pixel-level adversarial loss* L_{pix} for matching the distributions of two domains in low-level pixel-space; an *feature-level similarity loss* L_{fea} to guide model to capture high-level domain-invariant features; a *cycle consistency loss* L_{cyc} to prevent the learned bidirectional mappings G_{ST} and G_{TS} from contradicting each other [28]; and a *semantic consistency loss* L_{sem} that encourages the consistency of the key discriminative semantic contents during the pixel-level mapping across domains.

Pixel-level Adversarial Loss. The two generators are augmented by two adversarial discriminators respectively. A target discriminator D_T distinguishes between the real target data $\mathbf{x}_t$ and generated target data $G_{ST}(\mathbf{x}_s)$. In the same way, a source discriminator D_S distinguishes between the real source data $\mathbf{x}_s$ and the generated source data $G_{TS}(\mathbf{x}_t)$. Specifically, for the generator G_{ST}, it tries to map a source domain image to an adapted target domain sample $\mathbf{x}_t^g = G_{ST}(\mathbf{x}_s)$ that cannot be distinguished by its corresponding discriminator D_T , where the discriminator D_T is trained to do as well as possible in detecting generated "fake" target domain image $\mathbf{x}_t^g$. More formally, the generator $G_{ST}(\mathbf{x}_s)$ is trained with D_T by adversarial learning with the loss:

$$L_{adv}(G_{ST}, D_T, \mathbf{X}^s, \mathbf{X}^t) = E_{\mathbf{x}_t\sim\mathbf{X}^t}[\log(D_T(\mathbf{x}_t))] + E_{\mathbf{x}_s\sim\mathbf{X}^s}[\log(1 - D_T(G_{ST}(\mathbf{x}_s)))]. \tag{2}$$

Likewise, for the generator G_{TS} with the discriminator D_S, we introduce a similar adversarial learning process with the adversarial loss $L_{adv}\left(G_{TS}, D_S, \mathbf{X}^s, \mathbf{X}^t\right)$. The pixel-level adversarial

loss is defined as:

$$L_{pix} = L_{adv}(G_{ST}, D_T, \mathbf{X}^s, \mathbf{X}^t) + L_{adv}(G_{TS}, D_S, \mathbf{X}^s, \mathbf{X}^t). \tag{3}$$

Feature-level Similarity Loss. We also add a feature-level similarity loss to encourage that the high-level features from the adapted target images and the real target images are as similar as possible. The feature-level similarity loss L_{fea} is defined as Eq. 4 based on MMD [2], which is a kernel-based distance function widely used for the feature-level domain adaptation.

$$\begin{aligned} L_{fea}(C_T(G_{ST}(\mathbf{x}_s), C_T(\mathbf{x}_t))) &= ||E_{\mathbf{x}_s \sim \mathbf{X}^s}[\phi(C_T(G_{ST}(\mathbf{x}_s)))] - E_{\mathbf{x}_t \sim \mathbf{X}^t}[\phi(C_T(\mathbf{x}_t))]||^2 \\ &= E\left[K(C_T(G_{ST}(\mathbf{x}_s)), C_T(G_{ST}(\mathbf{x}_s)))\right] \\ &+ E\left[K(C_T(\mathbf{x}_t), C_T(\mathbf{x}_t))\right] \\ &- 2E\left[K(C_T(G_{ST}(\mathbf{x}_s)), C_T(\mathbf{x}_t))\right], \end{aligned} \tag{4}$$

where $K(\cdot, \cdot)$ denotes is a kernel function. In our experiments, we use a linear combination of multiple RBF kernels expressed as:

$$K(\mathbf{x}, \mathbf{y}) = \sum \eta_n \exp\left\{-\frac{1}{2\sigma_n}\|\mathbf{x} - \mathbf{y}\|^2\right\}, \tag{5}$$

where η_n and σ_n are the weight and the standard deviation for n-th RBF kernel [2], respectively.

Cycle Consistency Loss. Through the pixel level adversarial learning, ideally, G_{ST} could adapt the images from source domain to the images identically distributed as target domain. However, the adversarial loss alone still cannot guarantee that the contents of original samples could be reconstructed [28]. We hope that the image mapping from the source domain to the target domain should be a reversible process. In other word, the adapted image $G_{ST}(\mathbf{x}_s)$, which is generated by mapping a source domain image $\mathbf{x}_s$ to the target domain, should be able to back to the original image by the reversal mapping G_{TS}, that is $G_{TS}(G_{ST}(\mathbf{x}_s)) \approx \mathbf{x}_s$. Therefore, we impose a cycle-consistency constraint with L_1 normalization operator $\|\cdot\|_1$ as:

$$\begin{aligned} L_{cyc}(G_{ST}, G_{TS}, \mathbf{X}^s, \mathbf{X}^t) &= E_{\mathbf{x}_s \sim X^s}\left[\|G_{TS}(G_{ST}(\mathbf{x}_s)) - \mathbf{x}_s\|_1\right] \\ &+ E_{\mathbf{x}_t \sim \mathbf{X}^t}\left[\|G_{ST}(G_{TS}(\mathbf{x}_t)) - \mathbf{x}_t\|_1\right]. \end{aligned} \tag{6}$$

Semantic Consistency Loss. Although the cycle consistency loss in Eq. 6 can encourage the image mapping cycle to bring the source domain image back to the original image. There is no obvious constraint to ensure that the middle mapping could keep the semantic contents consistent. As shown in Fig. 1 (a), the mapping is free to shift the semantic contents, *i.e.* the image of class "3" may be transferred to the image of class "6".

To alleviate this issue, as illustrated in Fig. 1 (b), we enforce the middle mapping is semantic consistent. The basis of the semantic consistency is that the mapping from the labeled source domain to the target domain should keep the same class. To evaluate if the generated image $G_{ST}(\mathbf{x}_s)$ is at the same class with the source image $\mathbf{x}_s$, we introduce the pretrained source classifier C_s to do a preliminary inspection.

Given that the pretrained source classifier is noisy for the generated images, we use the output vector $C_S(\mathbf{x}_s)$ of source classier as a soft label vector to encourage that an image to be classified in the same way after mapping as it was before mapping. Due to our bidirectional pixel-level mappings are reversible, both the generated image and the reconstructed image should also keep the same semantics with the original image. Furthermore, we take full advantage of both soft label and hard label to augment semantic consistency during mapping processes, and the semantic consistency loss is defined as follows:

$$\begin{aligned} L_{sem}(G_{ST}, G_{TS}, \mathbf{X}^s, C_S) &= E_{\mathbf{x}_s \sim \mathbf{X}^s}\left[\|C_S(G_{ST}(\mathbf{x}_s)) - C_S(\mathbf{x}_s)\|^2\right] \\ &+ E_{\mathbf{x}_s \sim \mathbf{X}^s}\left[\|C_S(G_{TS}(G_{ST}(\mathbf{x}_s))) - C_S(\mathbf{x}_s)\|^2\right] \\ &+ L_{cls}(C_S, G_{TS}(G_{ST}(\mathbf{X}^s)), \mathbf{Y}^s). \end{aligned} \tag{7}$$

3.2 Optimization

As shown in Fig. 1 (b), the combination of objectives above will encourage a model to learn bidirectional pixel-to-pixel mappings between two domains, while keeping the same discriminative semantic content. By the discriminative pixel-to-pixel mapping from the source domain to the target domain, the generated target images $G_{ST}(\mathbf{x}_s)$ will preserve the label information from the source domain. Furthermore, a new target classifier C_T could be trained on the generated images as if trained on samples drawn from the target domain with minimizing the prediction loss:

$$L'_{cls}(C_T; G_{ST}(\mathbf{x}_s), \mathbf{Y}^s) = E_{(\mathbf{x}_s, \mathbf{y}_s)\sim(\mathbf{X}^s, \mathbf{Y}^s)} \left[-{\mathbf{y}_s}^\top \log(\sigma(C_T(G_{ST}(\mathbf{x}_s)))) \right]. \quad (8)$$

So far, G_{ST}, G_{ST}, D_S, D_T and C_T could be jointly optimized with the total optimization objective as:

$$L_{DA} = L'_{cls} + L_{pix} + L_{fea} + \lambda_{cyc} L_{cyc} + \lambda_s L_{sem} \quad (9)$$

where λ_{cyc} and λ_s are weights that control the interaction of losses to achieve better trade-off between the adaptation and classification. They are trained by an alternative training way in the concurrent sub-processes:

$$(\hat{\theta}_{G_{ST}}, \hat{\theta}_{G_{TS}}) = \underset{\theta_{G_{ST}}, \theta_{G_{TS}}}{\arg\min} \; L_{DA}, \quad (10)$$

$$(\hat{\theta}_{D_S}, \hat{\theta}_{D_T}) = \underset{\theta_{D_S}, \theta_{D_T}}{\arg\max} \; L_{pix}, \quad (11)$$

$$\hat{\theta}_{C_T} = \underset{\theta_{C_T}}{\arg\min} \; L'_{cls}. \quad (12)$$

where $\theta_{G_{ST}}$, $\theta_{G_{TS}}$, θ_{D_S}, θ_{D_T} and θ_{C_T} denote the parameters of the G_{ST}, G_{TS}, D_S, D_T and C_T respectively. The parameters can be updated by stochastic gradient descent optimization algorithms, like Adadelta [27].

4 Experiments

4.1 Datasets

We conduct experiments on 4 widely-used domain adaptation datasets: MNIST [12], USPS [10], MNIST-M [1], SVHN [19], as shown in Fig. 2. The statistics of the datasets are summarized in Table 1. For a fair comparison, we evaluate our algorithm on the 4 common domain adaptation tasks: MNIST $\rightarrow$ USPS (M $\rightarrow$ U), USPS $\rightarrow$ MNIST (U $\rightarrow$ M), MNIST $\rightarrow$ MNIST-M (M $\rightarrow$ M-M), SVHN $\rightarrow$ MNIST (S $\rightarrow$ M), using the training set only during training process and evaluating on the standard test sets. The token "$\rightarrow$" means the direction from the source domain to the target. The images are all resized to 28×28 pixels, and pixels of images are all normalized to $[0, 1]$. And we use grayscaled images for all tasks, except M $\rightarrow$ M-M task, where MNIST dataset were extended to three channels in order to match the shape of MNIST-M images (RGB images).

4.2 Experimental Setup

Network Architecture. Our network architecture is inspired by the CycleGAN [28]. The G_{ST} and G_{TS} use the same generative network architecture [28]. The generative network consists of 3 convolutional blocks, 9 residual blocks, and 3 transposed convolutional blocks. Each convolutional block consists of a convolutional layer followed by instance normalization layer and rectified linear unit (Relu) [18]. The architecture used for the discriminators D_S and D_T is a fully convolutional network with five convolutional layers. The networks used for the classifiers

Fig. 2: Dataset samples for our domain adaptation tasks.

Table 1: Datasets, "*/*" in columns of "Instances" denotes the number of train / test image pairs.

Dataset	Instances	classes	Image size	Color map
MNIST	60,000/10,000	10	28×28	Gray
USPS	7,291/2,007	10	28×28	Gray
MNIST-M	59,001/9,001	10	32×32	RGB
SVHN	73,257/26,032	10	32×32	RGB

C_S and C_T are composed of 4 convolutional layer followed by instance norm layer with leaky rectified linear unit (Leaky Relu) [15], 2 max-pooling layers, and a fully connected layer.

Training Details. All of our experiments are implemented with Tensorflow, and our implementation code will be released soon. We use the Adadelta optimizer [27] with a minibatch of size 16. Considering the regular adversarial loss suffers from the vanishing gradients problem, we replace the adversarial loss Eq. 3 with the least-squares GANs (LSGANs) loss [16], which can generate higher quality samples and perform more stable during the learning process.

Table 2: Accuracies ($mean \pm std$) on unsupervised domain adaptation among MNIST, USPS, SVHN and MNIST-M

Method	Reference	M→U	U→M	M→M-M	S→M
Source Only	ours	0.812	0.751	0.6070	0.6503
Target Only	ours	0.9729	0.9956	0.9545	0.9956
MMD	ICML 2015	0.8110	-	0.7690	0.7110
Domain Confusion	ICCV 2015	0.791±0.005	0.665±0.033	-	0.681±0.003
DSN w/MMD	NIPS 2016	-	-	0.8050	0.7220
CoGAN	NIPS 2016	0.912±0.008	0.891±0.0008	0.620	-
DSN w/DANN	NIPS 2016	0.913	-	0.8320	0.827
DANN	JMLR 2016	0.771±0.018	0.730±0.020	0.7666	0.7385
DRCN	ECCV 2016	0.918±0.0009	0.7367±0.0004	-	0.8197±0.0016
ADDA	CVPR 2017	0.894±0.0002	0.901±0.0008	-	0.760±0.0018
pixel-DA	CVPR 2017	0.959	-	0.982	-
CyCADA	ICML 2018	0.956±0.002	0.965±0.001	0.976±0.002	0.904±0.004
DIFA	CVPR 2018	0.923±0.001	0.910±0.004	0.924±0.001	0.897±0.002
Image2Image	CVPR 2018	0.925	0.908	0.916	0.847
RAAN	CVPR 2018	0.89	0.921	-	0.892
SBADA-GAN	CVPR 2018	**0.976**	0.950	**0.994**	0.761
BADA	Ours	0.9483±0.0008	**0.9689±0.0004**	0.9872±0.0005	**0.9254 ± 0.0012**
BADA without L_{fea}	Ours	0.9531±0.0006	0.9651±0.0019	0.9866±0.0003	0.8498±0.0061

4.3 Comparison with Existing Methods

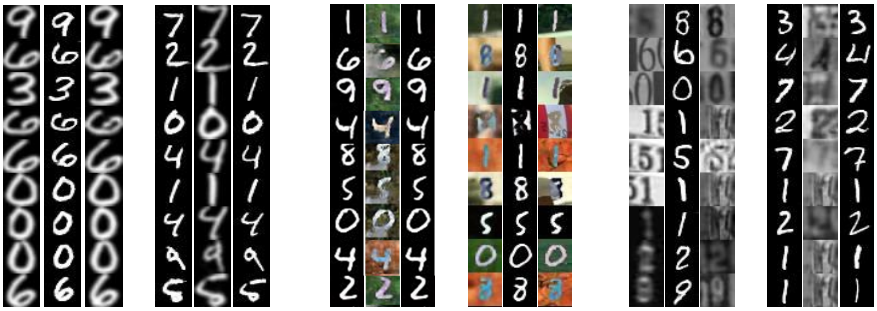
In this section, we compare the proposed BADA model with different domain adaptation (DA) methods among 4 widely adopted tasks. The compared methods are: (1) MMD [14,1], DSN w/MMD [2], Domain Confusion [24,25], DANN [5], DRCN [6], CoGAN [13], DSN w/DANN [2,1], ADDA [25], DIFA [26], and RAAN [3], which are feature-level DA methods; (2) pixel-DA [1], Image2Image [17], CyCADA [8] and SBADA-GAN [20], which are pixel-level DA methods. Table 2 presents the unsupervised DA recognition accuracy ($mean \pm std$) over three independent experiments. From Table 2, we can draw the follow observations:

- Firstly, we compare our BADA model with the "Source Only" and "Target Only" model. The "Source Only" and "Target Only" mean that the models are trained only on the source domain or target domain without any domain adaptation, respectively. They can be seen as a lower bound and an upper bound, respectively. We observe that our model achieves much better results than the "Source Only". It's more exciting that our results are much closer to the "Target Only".
- Compared with feature-level methods, our model not only achieves much better performance than MMD [14,1] and DSN w/MMD [2], which use traditional MMD loss [14,2] to minimize the feature-level difference between the source and target domain. But also our model is superior to Domain Confusion, DANN, CoGAN, DSNw/DANN, ADDA, DIFA and RAAN that are based on the feature-level adversarial method. This mainly owes to the proposed BADA model being able to capture the semantic contents transferred from the source domain to the target, by learning a bidirectional discriminative pixel-to-pixel mapping.
- Compared with pixel-level methods, our model outperforms the best competitor, pixel-DA on the M→M-M task, which is also an unsupervised pixel-level domain adaptation model with GAN. However, the pixelDA algorithm assumes that there are similar backgrounds between the source and target domain, which cannot perform well on more difficult S→M task. While our model outperforms the state-of-the-art CyCADA [8] model with a accuracy gap greater than 2.5% on the S→M task. This indicates the advantage of using the bidirectional pixel-level mapping with semantic consistency than the unidirectional pixel-level mapping with content similarity in pixelDA.
- Furthermore, the comparisons with CyCADA and SBADA-GAN also show the superiority of our bidirectional semantic consistency constraint. Although the SBADA-GAN method combines the source and target classifier for final prediction, which achieved the best performance on two tasks, our method outperforms it with accuracy gaps greater than 16.4% on the more difficult S→M task.

4.4 Evaluation on Semantic Consistency

Qualitative Analysis. In order to ensure that the proposed model could learn two semantic consistent mappings, we first visualize the bidirectional mapping results of the model in different tasks. As shown in Fig. 3, the proposed BADA learns a semantic consistent forward mapping from the source domain to the target with an inverted semantic consistent mapping simultaneously.

Quantitative Analysis. Furthermore, we demonstrate the quantitative analysis of the semantic consistency in Table 3. The first three rows represent the accuracy of original source image $\mathbf{x}_s$ on source classifier, generated target image $G_{ST}(\mathbf{x}_s)$ on the adapted target classifier C_T, and the reconstructed source image $G_{TS}(G_{ST}(\mathbf{x}_s))$ on the source classifier C_S, respectively. Accordingly, the last three rows report the accuracy of target image $\mathbf{x}_t$ on the adapted target classifier, generated source image $G_{TS}(\mathbf{x}_t)$ on the well-trained source classifier C_S, and the reconstructed target image $G_{ST}(G_{TS}(\mathbf{x}_t))$ on the target classifier C_T. We can observe that both the transferred and reconstructed images are recognizable by the corresponding classifiers, which can prove the semantic consistency during our dual pixel-to-pixel mappings. A comparison between



(a) U→M (b) M→M-M (c) S→M

Fig. 3: The visualization of pixel-to-pixel mapping: The left triple shows the mapping from the source domain to the target domain and back to the original source domain. The right triple shows the inverted mapping. Each triple consists of the original image(left), the generated image(middle), and the reconstructed image(right).

the $4th$ row and $5th$ rows in Table 3 shows that the performance of the adapted target images on the source classifier C_S could even nearly equal to the performance of the real target images on the target classifier. It indicates that the well-trained source classifier C_S can be shared with the target domain, while we only need to transfer the target image to the source image by the mapping we have learnt.

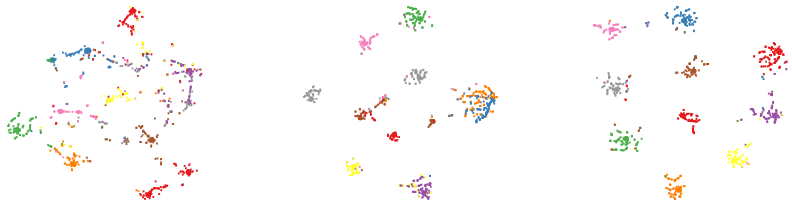
Table 3: Qualitative analysis of semantic consistency.

Method	M→U	U→M	M→M-M	S→M
$C_S(\mathbf{x}_s)$	0.9956	0.9729	0.9956	0.9308
$C_T(G_{ST}(\mathbf{x}_s))$	0.9821	0.9640	0.9902	0.8941
$C_S(G_{TS}(G_{ST}(\mathbf{x}_s)))$	0.9868	0.9670	0.9935	0.8721
$C_T(\mathbf{x}_t)$	0.9483	0.9689	0.9872	0.9254
$C_S(G_{TS}(\mathbf{x}_t))$	0.9550	0.9675	0.9907	0.9113
$C_T(G_{ST}(G_{TS}(\mathbf{x}_t)))$	0.9432	0.9663	0.9866	0.9008

4.5 Ablation Study

Effect of Feature-level Similarity Loss. The feature-level similarity loss L_{fea} is used to encourage the robustness of model. In order to investigate the effect of the feature-level similarity loss in more detail , we develop and evaluate two variations of BADA: BADA without L_{fea} and BADA, while keeping the optimization procedure in the same way. Table 2 shows the performance of two variations on the four widely adopted tasks. We can observe that BADA without L_{fea} has similar performances with BADA in different domain adaptation tasks, but one task on the S→M, where BADA performs much better. We infer that the pixel-level mapping combined with L_{fea} could capture more difficult domain shifts to get higher performance. Furthermore, we visualize the distribution of the target images in task S→M after training on source only and BADA using t-SNE tool respectively. A comparison between Fig. 4(a) and Fig. 4(b) reveals that

our semantic consistent pixel-level BADA without L_{fea} still has the ability to learn an adapted classifier for unsupervised target domain. Furthermore, as shown in Fig. 4(b) and Fig. 4(c), the proposed model combined with feature-level similarity loss further boosts the performance.



(a) Source only (b) BADA without L_{fea} (c) BADA

Fig. 4: The t-SNE visualizations of target domain samples features trained on (a) source only, (b) BADA without L_{fea}, (c) BADA with L_{fea} for the S→M task. We use 1000 test samples to generate the t-SNE plots.

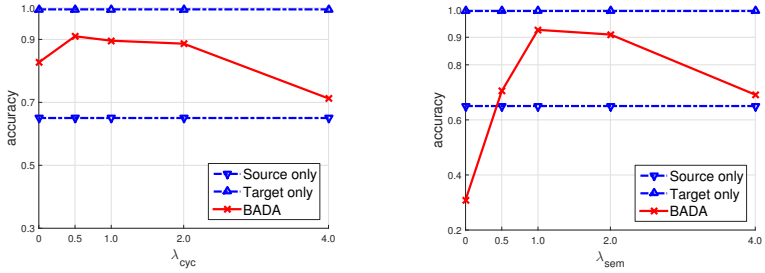
(a) Without L_{sem} (b) Without L_{cyc}

Fig. 5: The domain adaptation results of the proposed BADA without semantic consistency or without cycle consistency. In subfigures (a) and (b), a triple in each row consists of three images: i) *left* is the source SVHN image; ii) *middle* is the generated target MNIST image; and iii) *right* is the reconstructed source SVHN image.

Effect of consistency in BADA. In this scenario, we verify the importance of the *cycle consistency loss* L_{cyc} and *semantic consistency loss* L_{sem} for our pixel-to-pixel mapping. We developed and assessed two variations of our BADA: no semantic consistency or no semantic consistency, which mean BADA without L_{sem} or without L_{cyc}, respectively, while keeping the other loss satisfied and use the similar optimization. Figure 5 shows the results of the mapping from the source domain to the target domain, and back to the original source domain in pixel-level. When there is no semantic consistency but with cycle consistency, the mapping from the source domain to the target domain suffers the shift of semantic contents, despite the good reconstruction of the original images. Conversely, when there is no cycle consistency but with semantic consistency, the middle mapping could preserve the semantic contents, although, the reconstructed source images

are failed to be consistent with the original images. The two cases indicate that both the cycle consistency and semantic consistency contribute to the overall performance of model.

Parameter Sensitive Analysis. In this scenario, we evaluate the sensitiveness of the hyper-parameter λ_{cyc} and λ_{sem} on the performance of unsupervised domain adaptation. In the objective function Eq. 9, λ_{cyc} and λ_{sem} control the contributions of cycle consistency and semantic consistency respectively. Here, we conduct the experiments on the SVHN $\rightarrow$ MNIST task, where 2000 samples randomly selected from target test set as a validation set. Specifically, we explore the different λ_{cyc} and λ_{sem} from $0, 0.5, 1.0, 2.0, 4.0$. As aforementioned, $\lambda_{cyc} = 0$ and $\lambda_{sem} = 0$ indicate the proposed BADA without cycle consistency or without semantic consistency, respectively. The evaluation is conducted by changing one parameter (e.g. $Lcyc$) while keeping the other hyper-parameters fixed. As shown in Figure 6, both λ_{cyc} and λ_{sem} are important to the overall performance. Note that, when $\lambda_{sem} = 0$, the model performs badly. Thus it indicates that the λ_{sem} plays an essential role in the proposed model.



(a) BADA with different λ_{cyc}. (b) BADA with different λ_{sem}.

Fig. 6: Effect of model parameters (a) λ_{cyc} and (b) λ_{sem} in the proposed BADA.

5 Conclusion

In this paper, we proposed a novel BADA model to adapt the source domain images to appear as if drawn from the target domain by learning a pair of bidirectional pixel-level mappings that keep semantic consistency. BADA is capable to transfer the label information from the source domain to the target domain to learn a good target classifier, meanwhile it is advantaged to adapt the target images to the source domain to share the well-trained source classifier. Comprehensive experimental results on some widely used benchmark datasets show that the proposed BADA method outperforms the state-of-the-art domain adaptation methods with advances on superior visualization and semantic consistency analysis.

References

1. Bousmalis, K., Silberman, N., Dohan, D., Erhan, D., Krishnan, D.: Unsupervised pixel-level domain adaptation with generative adversarial networks. In: Proc.CVPR (2017)
2. Bousmalis, K., Trigeorgis, G., Silberman, N., Krishnan, D., Erhan, D.: Domain separation networks. In: Proc.NIPS (2016)
3. Chen, Q., Liu, Y., Wang, Z., Wassell, I., Chetty, K.: Re-weighted adversarial adaptation network for unsupervised domain adaptation. In: Proc.CVPR (2018)

4. Fernando, B., Habrard, A., Sebban, M., Tuytelaars, T.: Unsupervised visual domain adaptation using subspace alignment. In: Proc.ICCV (2013)
5. Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., Lempitsky, V.: Domain-adversarial training of neural networks. The Journal of Machine Learning Research **17**(1), 2096–2030 (2016)
6. Ghifary, M., Kleijn, W.B., Zhang, M., Balduzzi, D., Li, W.: Deep reconstruction-classification networks for unsupervised domain adaptation. In: Proc.ECCV (2016)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proc.CVPR (2016)
8. Hoffman, J., Tzeng, E., Park, T., Zhu, J.Y., Isola, P., Saenko, K., Efros, A., Darrell, T.: CyCADA: Cycle-consistent adversarial domain adaptation. In: Proc.ICML (2018)
9. Huang, G., Liu, Z., Weinberger, K.Q., van der Maaten, L.: Densely connected convolutional networks. In: Proc.CVPR. vol. 1, p. 3 (2017)
10. Hull, J.J.: A database for handwritten text recognition research. IEEE Transactions on pattern analysis and machine intelligence **16**(5), 550–554 (1994)
11. Kim, T., Cha, M., Kim, H., Lee, J.K., Kim, J.: Learning to discover cross-domain relations with generative adversarial networks. In: Proc.ICML (2017)
12. LeCun, Y., Cortes, C., Burges, C.: Mnist handwritten digit database. AT&T Labs [Online]. **2** (2010)
13. Liu, M.Y., Tuzel, O.: Coupled generative adversarial networks. In: Proc.NIPS (2016)
14. Long, M., Cao, Y., Wang, J., Jordan, M.I.: Learning transferable features with deep adaptation networks. arXiv preprint arXiv:1502.02791 (2015)
15. Maas, A.L., Hannun, A.Y., Ng, A.Y.: Rectifier nonlinearities improve neural network acoustic models. In: Proc. ICML (2013)
16. Mao, X., Li, Q., Xie, H., Lau, R.Y., Wang, Z., Smolley, S.P.: Least squares generative adversarial networks. In: Proc.ICCV. IEEE (2017)
17. Murez, Z., Kolouri, S., Kriegman, D., Ramamoorthi, R., Kim, K.: Image to image translation for domain adaptation. In: Proc.CVPR (2018)
18. Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: Proc.ICML (2010)
19. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y.: Reading digits in natural images with unsupervised feature learning. In: NIPS workshop on deep learning and unsupervised feature learning. vol. 2011, p. 5 (2011)
20. Russo, P., Carlucci, F.M., Tommasi, T., Caputo, B.: From source to target and back: Symmetric bi-directional adaptive gan. In: Proc.CVPR (2018)
21. Shrivastava, A., Pfister, T., Tuzel, O., Susskind, J., Wang, W., Webb, R.: Learning from simulated and unsupervised images through adversarial training. In: Proc.CVPR (2017)
22. Sun, B., Feng, J., Saenko, K.: Return of frustratingly easy domain adaptation. In: Proc.AAAI (2016)
23. Sun, B., Saenko, K.: Deep coral: Correlation alignment for deep domain adaptation. In: Proc.ECCV (2016)
24. Tzeng, E., Hoffman, J., Darrell, T., Saenko, K.: Simultaneous deep transfer across domains and tasks. In: Proc.ICCV (2015)
25. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. In: Proc.CVPR (2017)
26. Volpi, R., Morerio, P., Savarese, S., Murino, V.: Adversarial feature augmentation for unsupervised domain adaptation. In: Proc.CVPR (2018)
27. Zeiler, M.D.: Adadelta: an adaptive learning rate method. arXiv preprint arXiv:1212.5701 (2012)
28. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proc.CVPR (2017)
29. Zhuo, J., Wang, S., Zhang, W., Huang, Q.: Deep unsupervised convolutional domain adaptation. In: Proceedings of the 2017 ACM on Multimedia Conference. ACM (2017)