International Journal of Computer Vision (2021) 129:267–283
https://doi.org/10.1007/s11263-020-01364-5

Unsupervised Domain Adaptation in the Wild via Disentangling Representation Learning

Haoliang Li[1] · **Renjie Wan**[1] · **Shiqi Wang**[2] · **Alex C. Kot**[1]

Received: 19 December 2019 / Accepted: 29 July 2020 / Published online: 11 August 2020

Abstract
Most recently proposed unsupervised domain adaptation algorithms attempt to learn domain invariant features by confusing a domain classifier through adversarial training. In this paper, we argue that this may not be an optimal solution in the real-world setting (a.k.a. in the wild) as the difference in terms of label information between domains has been largely ignored. As labeled instances are not available in the target domain in unsupervised domain adaptation tasks, it is difficult to explicitly capture the label difference between domains. To address this issue, we propose to learn a disentangled latent representation based on implicit autoencoders. In particular, a latent representation is disentangled into a global code and a local code. The global code is capturing category information via an encoder with a prior, and the local code is transferable across domains, which captures the "style" related information via an implicit decoder. Experimental results on digit recognition, object recognition and semantic segmentation demonstrate the effectiveness of our proposed method.

Keywords In the wild · Cross-domain · Recognition · Segmentation

1 Introduction

Recently, though deep learning models have shown the great success on many computer vision applications, collecting sufficient labeled training data to train a deep model could be cumbersome. Domain adaptation (Pan and Yang 2010) aims to mitigate this problem by transferring the knowledge learned from a domain of rich labeled data to a new target domain of scarce annotation resource. Recently proposed unsupervised domain adaptation methods (Bousmalis et al. 2017; Tzeng et al. 2017) leverage the advantage of adversarial learning (Goodfellow et al. 2014) to learn domain invariant features. More recently, Liu et al. (2018) proposed to encode a shareable latent feature between the source domain and the target domain based on variational autoencoders (Kingma and Welling 2013). Hoffman et al. (2018) proposed to align both pixel level and feature level distributions in an adversarial learning manner.

A major drawback of most of the existing approaches is that, when aligning distribution between source and target domains, they only aim to align the data (e.g. feature, pixel) distribution while the category information is largely ignored. In the real world setting, the category information of different domains could be different (Schölkopf et al. 2012). For example, in a cross-domain recognition task, a target domain could only contain a portion of category information compared with the source domain as some digits may appear more frequently while others may not, according to Benford's Law (Benford 1938) (Fig. 1a). As another example, in a cross-domain semantic segmentation task, although the category information is maintained consistently between domains, the proportion of each category may be different (Fig. 1b). Based on our analysis on performing adaptation using CycleGAN (Zhu et al. 2017), we find that the target domain images generated by CycleGAN using source domain images may belong to some categories which are not in the label space of the

Communicated by Mei Chen, Cha Zhang and Katsushi Ikeuchi.

✉ Haoliang Li
lihaoliang@ntu.edu.sg

Renjie Wan
rjwan@ntu.edu.sg

Shiqi Wang
shiqwang@cityu.edu.hk

Alex C. Kot
eackot@ntu.edu.sg

[1] Rapid-Rich Object Search Lab, Nanyang Technological University, Singapore, Singapore

[2] Department of Computer Science, City University of Hong Kong, Kowloon, China

target domain. Therefore, we argue that, in order to obtain enhanced discriminative representations, category information should be taken into account.

In this paper, to tackle the problem of unsupervised domain adaptation in the wild, we propose a novel framework by leveraging the advantage of disentangling representation learning. The framework is built upon autoencoders, which have been widely adopted in deep learning based domain adaptation methods (Ghifary et al. 2016; Hoffman et al. 2018; Liu et al. 2017). Different from other disentanglement based domain adaptation methods (e.g. CDRD Liu et al. 2018 a state-of-the-art domain adaptation method based on conditional GANs), the conditional code which captures the category information is learned in an end-to-end manner instead of being randomly assigned, avoiding the negative transfer in an efficient manner. In particular, we take both categories and the non-category style information into consideration, and build our model upon implicit autoencoders (Makhzani 2018). The latent category information is encoded with the regularization of category distribution, and the remaining information is captured by the implicit conditional likelihood distribution based on conditional GANs, with the assumption that the source domain and the target domain data share a common latent space regarding the "style" related (a.k.a. non-category) information while the category information can be largely different. We evaluate our proposed framework on three different vision tasks: digit recognition, object recognition and semantic segmentation. Experimental results show that the proposed method can achieve significantly better performance compared with other state-of-the-art unsupervised domain adaptation methods.

2 Related Works

2.1 Unsupervised Domain Adaptation

Besides the traditional domain adaptation approaches based on either subspace learning (Pan et al. 2011) or instance reweighting (Huang et al. 2006), deep learning based methods have also been proved to be effective for domain adaptation. Ghifary et al. (2016) showed that the transferable capability can be improved with reconstruction loss. The Deep Adaptation Network (Long et al. 2015) applied Maximum Mean Discrepancy (MMD) to multiple layers based on AlexNet (Krizhevsky et al. 2012), where network parameters as well as the parameters for the RBF kernel are jointly optimized to obtain a suitable distance measurement in a Reproduced Kernel Hilbert Space (RKHS). Long et al. (2016) further showed that introducing addition residual block can further improve the performance by learning more domain-invariant feature representation. More recently, Pan et al. (2019) proposed to extend MMD by prototypical network embedding for domain adaptation. Besides MMD, Haeusser et al. (2017) proposed a novel discrepancy measure by considering label information. Moreover, adversarial learning can also benefit domain adaptation. In Ganin et al. (2016), a domain classifier was introduced to assign a binary label to either source or target domain, where the domain classifier was encouraged that its prediction was close to a uniform distribution of both source and target domains. The gradient reversal algorithm (ReverseGrad) also introduced a domain classifier by maximizing the domain loss directly. It was also reported that simply adopting data augmentation with self-ensemble learning can also benefit domain adaptation (French et al. 2017). As generative adversarial network (GAN) is equivalent to minimize JS divergence (Goodfellow et al. 2014), several works also leverage GAN and its extension CycleGAN (Zhu et al. 2017) for domain adaptation (Bousmalis et al. 2017; Hoffman et al. 2018; Russo et al. 2018; Tzeng et al. 2017), which can be further applied to classifier boundary level (Saito et al. 2018). Recently, Liu et al. (2018) proposed a domain adaptation method which also aimed at learning disentangled feature representation. In particular, the common latent representation between source and target domains was encoded through variational autoencoder while the attribute information was represented by a random distribution regularized by an auxiliary classifier to maximize the mutual information between random distribution and the synthesized images, which is known as infoGAN (Chen et al. 2016). However, the authors in Liu et al. (2018) proposed to capture the categorical information only by randomly assigning one-hot vector as the input based on conditional GAN, which may fail to handle the imbalanced category situation for unsupervised domain adaptation problem. Different from Liu et al. (2018), we aim to conduct disentanglement by learning the category information in an end-to-end learning manner, where category information can be captured by matching it with a prior distribution instead of randomly assigning one-hot vector. It is worth mentioning that there are several works which targeted on partial domain adaptation (e.g. Cao et al. 2019). However, compared with these approaches, our proposed method is more general as the existing techniques based on partial domain adaptation only considered object recognition task for evaluation.

2.2 Image-to-Image Translation

The problem of image-to-image translation has attracted more and more attentions due to the success of GAN. Isola et al. (2017) proposed the first framework for small-scale image-to-image translation problem based on conditional GAN, which has been further extended to high-resolution image in Wang et al. (2017). Besides conditional GAN, recent works also focus on conducting image translation in an unsu-

(a) Digit recognition **(b)** Semantic segmentation

Fig. 1 Illustrations of our motivation for the proposed unsupervised domain adaptation in the wild, where imbalanced category settings are observed. The first and second rows show the samples from source and target domains, respectively. First row: according to the Benford's Law, a target domain can only contain a portion of category information compared with the source domain. Second row: although the category information is similar between source and target domain, the proportion of each category may be different. For example, the trees in the first row are in the middle and right part of the image while the trees are mainly in the left for the image in the second row, which leads to category inconsistency in pixel level

pervised manner by leveraging pixel values (Shrivastava et al. 2017), pixel gradients (Bousmalis et al. 2017) and semantic features (Taigman et al. 2016).

Image-to-image translation problem is closely related to unsupervised domain adaptation, as it can be treated as aligning distribution divergence between source and target domains. Bousmalis et al. (2017) also conducted experiments on domain adaptation by adapting source-domain images to appear as if drawn from the target domain. In addition, coupled GAN (Liu and Tuzel 2016) and its extension UNIT (Liu and Tuzel 2016) further assumed that a shared latent space exists such that the corresponding images from two domains can be mapped to the same latent code besides pixel level adaptation. To improve the diversity of translated images, CycleGAN (Zhu et al. 2017) was proposed and further applied to unsupervised domain adaptation problem (Hoffman et al. 2018). Recently, Huang et al. (2018) leveraged the advantage of disentangled representation learning by proposing a multimodel unsupervised image-to-image framework. Our proposed method also leverages the advantage of image-to-image translation for domain adaptation.

2.3 Cross-Domain Semantic Segmentation

To the best of our knowledge, the first work tackling the cross domain segmentation problem was introduced in Levinkov and Fritz (2013) with the scene prior based on the Bayesian model. To reduce the domain shift in the segmentation task, Hoffman et al. (2016) proposed an adversarial based approach to align the scene distinction between simulated and real environments. A number of works further extended the idea by considering adaptation based on semantic feature (Chen et al. 2018; Saito et al. 2018), output layout (Tsai et al. 2018; Vu et al. 2019), superpixels (Zhang et al. 2017) and translated images (Sankaranarayanan et al. 2017; Zhang et al. 2018). In Hoffman et al. (2018), cross-domain segmentation was conducted by considering adaptation based on semantic level and pixel level. Our motivations in tackling this problem are twofold. First, the domain shift originates from different style rendering, such that we aim to conduct disentangling representation by assuming that there exists a space which captures the style information. Secondly, to ensure that meaningful category information can be captured through disentangling representation learning, the category information is further used for image reconstruction and translation based on conditional GAN (Wang et al. 2017).

3 Preliminaries

Before introducing our proposed framework for unsupervised domain adaptation, we first revisit implicit autoencoders (Makhzani 2018), which our proposed framework is built upon and can be regarded as extensions of adversarial autoencoders (Makhzani et al. 2015) to perform variational inference by matching the aggregated posterior of the latent code with an arbitrary prior in an adversarial training manner. Specifically, let $\mathbf{x}$ be a data point which is drawn from a distribution $p(\mathbf{x})$. The encoder defines a posterior distribution $q(\mathbf{z}|\mathbf{x})$ which maps data $\mathbf{x}$ to the latent vector $\mathbf{z}$. The decoder defines a conditional distribution $p(\mathbf{x}|\mathbf{z})$ to output a reconstructed data point $\hat{\mathbf{x}}$. Therefore, the aggregated posterior distribution $q(\mathbf{z})$ and the reconstruction distribution $r(\hat{\mathbf{x}})$ can be defined as

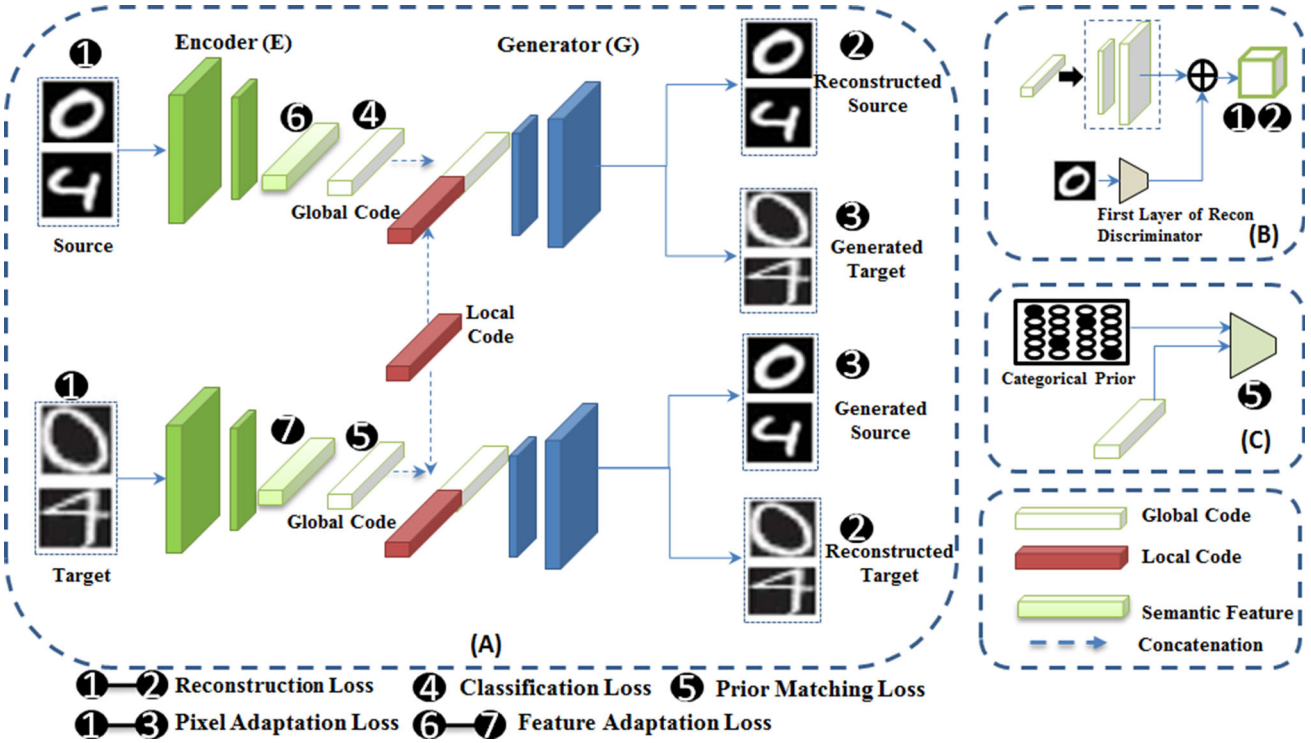

Fig. 2 The proposed domain adaptation framework. **a** Implicit autoencoder framework where the parameters of encoder and generator are shared by source and target domains (the discriminators are omitted for better visualization). **b** Conditional network for reconstruction GAN (Makhzani 2018; Makhzani and Frey 2017). **c** Adversarial loss for prior matching

$$q(\mathbf{z}) = \int_{\mathbf{x}} q(\mathbf{x}, \mathbf{z}) d\mathbf{x} = \int_{\mathbf{x}} q(\mathbf{z}|\mathbf{x}) p(\mathbf{x}) d\mathbf{x},$$
$$r(\hat{\mathbf{x}}) = \int_{\mathbf{z}} r(\hat{\mathbf{x}}, \mathbf{z}) d\mathbf{z} = \int_{\mathbf{z}} q(\mathbf{z}) p(\mathbf{x}|\mathbf{z}) d\mathbf{z}, \quad (1)$$

where $q(\mathbf{z})$ is regularized by a predefined prior $p(\mathbf{z})$ (categorical prior in our case) in an adversarial manner,

$$\mathcal{L}_{prior} = \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})}[\log(D_{prior}(\mathbf{z}))] + \mathbb{E}_{\mathbf{z}\sim q(\mathbf{z})}[\log(1 - D_{prior}(\mathbf{z}))]. \quad (2)$$

An implicit autoencoder is different from an adversarial autoencoder in two aspects. First, instead of learning a decoder based on the aggregated latent code, the implicit autoencoder learns a decoder based on the conditional GAN, which takes another prior as the conditional extra information. Second, the implicit autoencoder proposes adversarial reconstruction instead of the deterministic L_1/L_2 reconstruction loss adopted by other autoencoders frameworks (e.g. Makhzani et al. 2015; Ngiam et al. 2011). The adversarial reconstruction loss is given by

$$\mathcal{L}_{recon} = \mathbb{E}_{\mathbf{x},\mathbf{z}\sim q(\mathbf{x},\mathbf{z})}[\log(D_{recon}(\mathbf{x}, \mathbf{z}))] + \mathbb{E}_{\hat{\mathbf{x}},\mathbf{z}\sim r(\hat{\mathbf{x}},\mathbf{z})}[\log(1 - D_{recon}(\hat{\mathbf{x}}, \mathbf{z}))]. \quad (3)$$

4 Proposed Methodology

Recently proposed deep learning based domain adaptation methods (Hoffman et al. 2018; Liu et al. 2017; Tzeng et al. 2015, 2017) share a common assumption that there exists a latent code which can capture the domain invariant information between the source and the target domains. However, as shown in Ming Harry Hsu et al. (2015), directly conducting domain adaptation based on feature level may lead to negative transfer, as the category information is ignored. To mitigate this problem, we propose to decompose the latent code into two parts in an end-to-end learning manner: one is to capture the category information, which can be different across domains, and the other is to capture the transferable information. In particular, we extend implicit autoencoders (Makhzani 2018) to the unsupervised domain adaptation setting. We aim to train a classifier based on the encoder. To encourage the output of encoder to capture the discrete category information, we introduce a prior distribution to regularize the latent code, and a generator based on the decoder to further impose reconstruction constraints for cross-domain image translation.

4.1 Framework

Our proposed algorithm is built upon the implicit autoencoders introduced in Sect. 3. We illustrate the whole framework in Fig. 2. The whole architecture consists of six subnetworks: one encoder E, one generator G and four discriminators D_{prior}, D_{pixel}, D_{recon} and D_{feat}. In our domain adaptation setting, we have the source domain images $\mathbf{X}_S = \{\mathbf{x}_S\}$'s and the target domain images $\mathbf{X}_T = \{\mathbf{x}_T\}$'s, where $\mathbf{x}_S, \mathbf{x}_T \in \mathbb{R}^{H\times W\times C}$ with H and W denoting the height and width of an image respectively, and C is the size of input channel. The encoder $E = E_c \circ E_f$ acts as a combination of a feature mapping E_f and a classifier E_c by mapping the input $\mathbf{x}_S$ or $\mathbf{x}_T$ to a semantic feature representation $E_f(\mathbf{x}_S)$ or $E_f(\mathbf{x}_T)$ and a softmax global code $E(\mathbf{x}_S)$ or $E(\mathbf{x}_T)$. The generator G with the output size $\mathbb{R}^{H\times W\times 2C}$ defines an implicit conditional model distribution for the image reconstruction and the image generation purposes, which can be defined as $[\hat{\mathbf{x}}_S, \mathbf{x}_{S\to T}] = G(E(\mathbf{x}_S), \mathbf{n})$, $[\mathbf{x}_{T\to S}, \hat{\mathbf{x}}_T] = G(E(\mathbf{x}_T), \mathbf{n})$, where $\mathbf{n}$ is the local code randomly generated based on a Gaussian distribution and is shared by both the source and the target domains, $\hat{\mathbf{x}}_S, \hat{\mathbf{x}}_T \in \mathbb{R}^{H\times W\times C}$ are the reconstructed images of $\mathbf{x}_S, \mathbf{x}_T$, respectively. $\mathbf{x}_{S\to T}, \mathbf{x}_{T\to S} \in \mathbb{R}^{H\times W\times C}$, $\mathbf{x}_{S\to T} \in \mathbf{X}_{S\to T}$, $\mathbf{x}_{T\to S} \in \mathbf{X}_{T\to S}$ are the translated images based on $\mathbf{x}_S, \mathbf{x}_T$, respectively. In our setting, the first C channels of the output of G correspond to the source domain and the last C channels correspond to the target domain. To implement how the reconstruction GAN conditions on the global code, we adopt a conditional network where the global code is used as the input as in Makhzani (2018) and Makhzani and Frey (2017), and the output is added to the first layer of the discriminator as an adaptive bias.

4.2 Implicit Autoencoders with Domain Adaptation

We then introduce the domain adaptation formulation. As our encoder directly maps input to the softmax global code, we impose a classification loss $\mathcal{L}_{class}$ supervised by groundtruth label/annotation $\mathbf{Y}_S$ based on the source domain. As we aim to conduct latent feature disentangling where the category information is captured by the encoder, we directly impose the standard cross-entropy loss based on the output of encoder.

For the domain adaptation loss, in analogous to (Hoffman et al. 2018), we also consider feature level adaptation as well as pixel level adaptation. Conducting feature level adaptation in an adversarial manner was proved to be effective for unsupervised domain adaptation. However, as the category information between the source and the target domains can be imbalanced, directly aligning latent feature representation may lead to negative transfer. We propose to disentangle the latent feature to category related code (a.k.a. global code) as well as style related code (a.k.a. local code) to address this problem. In particular, we first regularize the global code based on the target domain as

$$\begin{aligned}\mathcal{L}_{prior} &= \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z}_T)}[\log(D_{prior}(\mathbf{z}))] \\ &\quad + \mathbb{E}_{\mathbf{x}\sim\mathbf{X}_T}[\log(1 - D_{prior}(E(\mathbf{x})))],\end{aligned} \tag{4}$$

where D_{prior} is the discriminator defined in Sect. 4.1 by encouraging the posterior to match the prior. We only regularize the global code in the target domain as the source domain global code can be learned with cross-entropy loss. We define $p(\mathbf{z}_T)$ as categorical distribution prior based on the target domain which can help the latent global code to capture discrete factors of variation. The details regarding how to construct $p(\mathbf{z}_T)$ are provided in the experimental section. To capture the transferable information, we propose a shareable local code $\mathbf{n}$ between source and target domains which is generated by random Gaussian distribution and for feature level adaptation in non-categorical level. Thus, the disentangling of latent feature representation can be achieved. The local code $\mathbf{n}$ will be conditioned on the global code as the input to the generator G. In addition, as suggested in Hoffman et al. (2018) and Tzeng et al. (2017), we also conduct semantic feature adaptation between the source and the target domains, which leads to an additional feature level adversarial loss

$$\begin{aligned}\mathcal{L}_{feat} &= \mathbb{E}_{\mathbf{x}\sim\mathbf{X}_T}[\log(D_{feat}(E_f(\mathbf{x})))] \\ &\quad + \mathbb{E}_{\mathbf{x}\sim\mathbf{X}_S}[\log(1 - D_{feat}(E_f(\mathbf{x})))].\end{aligned} \tag{5}$$

For pixel level adaptation, we introduce a generator G defined in Sect. 4.1 as a mapping from source to target/target to source as $\mathbf{x}_{S\to T}$ and $\mathbf{x}_{T\to S}$ as well as a discriminator D_{pixel} for distinguishing purpose.

$$\begin{aligned}\mathcal{L}_{pixel} &= \mathbb{E}_{\mathbf{x}_S\sim\mathbf{X}_S}[\log(D_{pixel}(\mathbf{x}_S))] \\ &\quad + \mathbb{E}_{\mathbf{x}_{T\to S}\sim\mathbf{X}_{T\to S}}[\log(1 - D_{pixel}(\mathbf{x}_{T\to S}))] \\ &\quad + \mathbb{E}_{\mathbf{x}_T\sim\mathbf{X}_T}[\log(D_{pixel}(\mathbf{x}_T))] \\ &\quad + \mathbb{E}_{\mathbf{x}_{S\to T}\sim\mathbf{X}_{S\to T}}[\log(1 - D_{pixel}(\mathbf{x}_{S\to T}))].\end{aligned} \tag{6}$$

4.3 Model Training

By considering the objectives together, the loss function of our proposed framework is defined as

$$\begin{aligned}\mathcal{L} &= \lambda_0\mathcal{L}_{class} + \lambda_1\mathcal{L}_{prior} + \lambda_2\mathcal{L}_{feat} \\ &\quad + \lambda_3\mathcal{L}_{pixel} + \lambda_4\mathcal{L}_{recon},\end{aligned} \tag{7}$$

where the first term serves as classification purpose with cross-entropy loss on the source domain labeled data, the second is to capture prior information, the third and fourth term are for semantic feature- and pixel- level domain adaptation, respectively, and the last term is the stochastic reconstruction loss induced by the autoencoder based on both the source and

the target domains. For implementation, we follow (Goodfellow et al. 2014) to train the encoder by maximizing $\log D(E(\mathbf{x}))$ instead of minimizing $\log(1 - D(E(\mathbf{x})))$, as we find it can lead to more stable training.

Similar to Hoffman et al. (2018), a two-stage training process is conducted which can help with reducing the GPU memory cost and potential for volatile training. In particular, we first train with classification loss and the semantic feature adaptation loss, and then train the model with (7) by removing the semantic feature adaptation loss. During each training iteration, we forward the source image batch $\{\mathbf{x}_S\} \in \mathbf{X}_S$ with its corresponding groundtruth $\{\mathbf{y}_S\} \in \mathbf{Y}_S$, as well as the target image batch $\{\mathbf{x}_T\} \in \mathbf{X}_T$ to our model. The model is trained in an adversarial manner. We also discuss the performance by considering jointly training of the proposed objective in the experimental section. The whole process is summarized in Algorithm 1.

Algorithm 1 Unsupervised Domain Adaptation in the Wild with Implicit Autoencoders

Input: $\mathbf{X}_S$, $\mathbf{X}_T$ and $\mathbf{Y}_S$, initialized parameters E, G, D_{recon}, D_{prior}, D_{feat} and D_{pixel}.
Output: Learned parameters E^*, G^*, D^*_{recon}, D^*_{prior}, D^*_{feat} and D^*_{pixel}.
%: First Stage
while Stopping criterion is not met **do**
1: Sample image batch $\{\mathbf{x}_S\}$ and $\{\mathbf{x}_T\}$ from $\mathbf{X}_S$ and $\mathbf{X}_T$, respectively.
2: Compute the gradient of $\lambda_0\mathcal{L}_{class} + \lambda_2\mathcal{L}_{feat}$ w.r.t. D_{feat}.
3: Take a gradient step to update D_{feat} to maximize the objective in step 2 of first stage.
4: Compute the gradient of objective in step 2 of first stage w.r.t. E.
3: Take a gradient step to update E to to minimize the objective in step 2 of first stage.
end while
%: Second Stage
while Stopping criterion is not met **do**
1: Sample image batch $\{\mathbf{x}_S\}$ and $\{\mathbf{x}_T\}$ from $\mathbf{X}_S$ and $\mathbf{X}_T$, respectively. Randomly generate Gaussian noise $\mathbf{n}$. Generate categorical distribution $\mathbf{z}_T$ with the same size as the output of E.
2: Compute the gradient of $\lambda_0\mathcal{L}_{class} + \lambda_1\mathcal{L}_{prior} + \lambda_3\mathcal{L}_{pixel} + \lambda_4\mathcal{L}_{recon}$ w.r.t. D_{recon}, D_{prior} and D_{pixel}.
3: Take a gradient step to update D_{recon}, D_{prior} and D_{pixel} to maximize the objective in step 2 of second stage.
4: Compute the gradient of the objective in step 2 of second stage w.r.t. E and G.
5: Take a gradient step to update E and G to minimize the objective in step 2 of second stage.
end while

4.4 Theoretical Analysis

We give an explanation of our proposed methodology in the perspective of transfer learning theory in Ben-David et al. (2010).

Theorem 1 *Let $\theta \in \mathcal{H}$ be a hypothesis, $\epsilon_1(\theta)$ and $\epsilon_2(\theta)$ be the expected risks of domain 1 and 2 respectively, then*

$$\epsilon_2(\theta) \leq \epsilon_1(\theta) + 2d_{\mathcal{H}}(S, T) + Const \tag{8}$$

where $d_{\mathcal{H}}(S, T)$ is to measure the distribution divergence between source and target domain (Ben-David et al. 2010).

We further show that our proposed objective is equivalent to minimize the upper bound in Theorem 1.

Theorem 2 *Let $(\mathbf{x}_S, \mathbf{z}_S) \sim \mathcal{S}$, $(\mathbf{x}_T, \mathbf{z}_T) \sim \mathcal{T}$, the empirical risk based on target domain can be represented as shown below*

$$\begin{aligned}\epsilon_2(\theta) = \mathbb{E}_{\mathcal{T}}[-\log q(\mathbf{z}_T|\mathbf{x}_T)] \leq \mathcal{L}_{class} + \mathcal{L}_{prior} \\ + \frac{1}{2}(\mathcal{L}_{pixel} + \mathcal{L}_{recon}) + Const\end{aligned} \tag{9}$$

where $d_{\mathcal{H}}(S, T) = \frac{1}{2}\mathcal{L}_{prior} + \frac{1}{4}(\mathcal{L}_{pixel} + \mathcal{L}_{recon})$. $\mathcal{L}_{class}$ is the classification loss on source domain, which is $\epsilon_1(\theta)$ in Eq. 8, $\mathcal{L}_{prior}$ is the regularization loss by matching the latent categorical information with the prior distribution, $\mathcal{L}_{pixel}$ is the adversarial loss between source and target domain data, $\mathcal{L}_{recon}$ is stochastic reconstruction loss and $Const = \mathbb{E}_{S,T}[|\log \frac{p(\mathbf{x}_S)}{p(\mathbf{x}_T)}|]$, which is a constant determined by the input data.

To prove the theorem, we assume the joint distribution of a batch from a domain can approximate the joint distribution of its ideal domain, which yield $q(\mathbf{x}_S, \mathbf{z}_S) = p(\mathbf{x}_S, \mathbf{z}_S)$ and $q(\mathbf{x}_T, \mathbf{z}_T) = p(\mathbf{x}_T, \mathbf{z}_T)$. Such assumption has been adopted in many deep learning based domain adaptation techniques, which aimed to match the distribution of minibatch instead of the whole source and target domain. The proof is given below.

Proof

$$\begin{aligned}
&\mathbb{E}_{\mathcal{T}}[-\log q(\mathbf{z}_T|\mathbf{x}_T)] = \mathbb{E}_{\mathcal{T}}[|-\log q(\mathbf{z}_T|\mathbf{x}_T)|] \\
&\leq \mathbb{E}_{\mathcal{S},\mathcal{T}}[|-\log q(\mathbf{z}_T|\mathbf{x}_T) + \log q(\mathbf{z}_S|\mathbf{x}_S)|] \\
&+ \mathbb{E}_{\mathcal{S}}[|-\log q(\mathbf{z}_S|\mathbf{x}_S)|] \\
&\leq \mathbb{E}_{\mathcal{S},\mathcal{T}}[|-\log p(\mathbf{x}_T|\mathbf{z}_T) + \log p(\mathbf{x}_S|\mathbf{z}_S) - \log q(\mathbf{z}_T) \\
&\quad + \log q(\mathbf{z}_S)|] + \mathbb{E}_{\mathcal{S}}[|-\log q(\mathbf{z}_S|\mathbf{x}_S)|] \\
&+ \mathbb{E}_{\mathcal{S},\mathcal{T}}\left[\left|\log \frac{p(\mathbf{x}_S)}{p(\mathbf{x}_T)}\right|\right] \\
&\leq \mathbb{E}_{\mathcal{S},\mathcal{T}}\left[\left|\frac{1}{2}\log p(\mathbf{x}_T|\mathbf{z}_T) - \frac{1}{2}\log p(\mathbf{x}_S)\right|\right] \\
&\quad + \mathbb{E}_{\mathcal{S},\mathcal{T}}\left[\left|\frac{1}{2}\log p(\mathbf{x}_T|\mathbf{z}_T) - \frac{1}{2}\log p(\mathbf{x}_T)\right|\right] \\
&\quad + \mathbb{E}_{\mathcal{S},\mathcal{T}}\left[\left|\frac{1}{2}\log p(\mathbf{x}_S|\mathbf{z}_S) - \frac{1}{2}\log p(\mathbf{x}_S)\right|\right]
\end{aligned}$$

$$
\begin{aligned}
&+\mathbb{E}_{\mathcal{S},\mathcal{T}}\left[\left|\frac{1}{2}\log p(\mathbf{x}_S|\mathbf{z}_S)-\frac{1}{2}\log p(\mathbf{x}_T)\right|\right]\\
&+\mathbb{E}_{\mathcal{S}}[|-\log q(\mathbf{z}_S|\mathbf{x}_S)|]+\mathbb{E}_{S,T}\left[\left|\log\frac{p(\mathbf{x}_S)}{p(\mathbf{x}_T)}\right|\right]\\
&+\mathbb{E}_{\mathcal{S},\mathcal{T}}[|\log q(\mathbf{z}_S)-\log q(\mathbf{z}_T)-\log p(\mathbf{z})+\log p(\mathbf{z})|]\\
\leq&\underbrace{\frac{1}{2}(dist(p(\mathbf{x}_S),r(\hat{\mathbf{x}}_T))+dist(p(\mathbf{x}_T),r(\hat{\mathbf{x}}_S)))}_{cross\quad domain\quad loss}\\
&+\underbrace{\frac{1}{2}(dist(p(\mathbf{x}_S),r(\hat{\mathbf{x}}_S))+dist(p(\mathbf{x}_T),r(\hat{\mathbf{x}}_T)))}_{reconstruction\quad loss}\\
&+\underbrace{dist(q(\mathbf{z}_S),\,p(\mathbf{z}))+dist(q(\mathbf{z}_T),\,p(\mathbf{z}))}_{prior\quad loss}\\
&+\underbrace{\mathbb{E}_{\mathcal{S}}[|-\log q(\mathbf{z}_S|\mathbf{x}_S)|]}_{classification\quad loss}+\mathbb{E}_{S,T}\left[\left|\log\frac{p(\mathbf{x}_S)}{p(\mathbf{x}_T)}\right|\right]
\end{aligned}
$$

As $q(\mathbf{z}_S)$ is supervised by label information from the source domain, $dist(q(\mathbf{z}_S),\,p(\mathbf{z}))$ is also minimized. This completes the proof. □

In summary, our proposed method can be treated as minimizing the empirical risk on source domain as well as aligning the joint distribution of source and target domain on both pixel level and category level. While the pixel level adaptation is achieved by $\mathcal{L}_{pixel}$, the category level adaptation is jointly determined by matching the latent code with a Categorical prior, as well as stochastic reconstruction. The Categorical prior mainly aims to capture category information, and stochastic reconstruction further encourage the output of encoder to capture reliable categorical information, which can be used for reconstruction purpose. Surprisingly, we notice that the upper bound only depends the second phase of training. We show in our experimental study that removing semantic feature adaptation term has less impact to the final performance compared with other terms. However, we experimentally find that semantic feature adaptation term can help to train the model stably.

5 Experiment

We evaluate our proposed method for unsupervised domain adaptation in the wild on location-invariant and location-dependent tasks. For the first one, the cross-domain digit recognition and object recognition are considered. For the latter one, we focus on the semantic segmentation task, which aims to adapt the learned model based on synthetic dataset to the real-world one.

5.1 Cross-Domain Digit Recognition

5.1.1 Setting of Experiment

We first evaluate our method on cross-domain digit recognition by considering three different datasets, MNIST (LeCun et al. 1998), USPS (Hull 1994) and SVHN (Netzer et al. 2011). In particular, we consider two scenarios, (1) domain shift between different style grayscale images, (2) domain shift between RGB and grayscale images. We show some examples of MNIST, USPS and SVHN in Fig. 3.

5.1.2 MNIST ↔ USPS

To evaluate the effectiveness of our proposed methodology, we conduct experiments by considering the balanced setting, where all category information is available in the target domain, as well as the imbalanced setting, where the category information in the target domain is only a subset of the category information in the source domain. We adopt a LeNet like encoder in order to fairly compare with other baseline methods. Moreover, for the imbalanced setting, we choose to compare with four different baseline methods which conducted domain adaptation from different perspectives, ADDA (Tzeng et al. 2017) for feature level adaptation, MCD (Saito et al. 2018) for boundary level adaptation, CDRD (Liu et al. 2018), CYCADA (Hoffman et al. 2018) based on both feature level and pixel level and ETN (Pan et al. 2019), which is a recent proposed imbalanced domain adaptation algorithm.[1] We conduct the experiments for ten times and the average performances are reported for our proposed method. For the baselines, we report the best results obtained from the published papers. For imbalanced setting, we report their best results on the test data by varying their parameters in a wide range.

We set the batch size to 128, and follow the protocol in Bousmalis et al. (2017) and Russo et al. (2018) for parameter setting, which leads to $\lambda_0 = \lambda_2 = 1, \lambda_1 = 2, \lambda_3 = \lambda_4 = 10$. We use Stochastic Gradient Descend (SGD) as suggested in Makhzani and Frey (2017) to optimize our model and set the learning rate as 0.1 for encoder, generator as well as discriminator. Regarding Categorical distribution generation, we first compute the pseudo-label of target domain data based on the network trained by classification loss and semantic feature adaptation loss, and compute the histogram density h_k (where $k = \{0, 1, 2, \ldots, 9\}$ for digit recognition) based on the predicted label. We set the probability p_k of Categorical

[1] We omit other baseline methods under this setting as they can be categorized into the aforementioned baselines and achieved poorer performance.

Fig. 3 Examples of MNIST, USPS and SVHN datasets

Table 1 Network architectures used for MNIST ↔ USPS

Encoder	Generator	Disc. prior	Disc. feat
Conv5, maps 20	Linear 1024, BN 1024	Linear 100, ReLU	Linear 500, ReLU
Max Pool 2, Stride 2, ReLU	Linear 6272, BN 6272	Linear 100, ReLU	Linear 500, ReLU
Conv5, maps 50, Dropout(0.5)	DeConv4, Stride 2, maps 64, BN 64, ReLU	Linear 1	Linear 1
Max Pool 2, Stride 2, ReLU	DeConv4, Stride 2, maps 2, tanh		
Linear 500, ReLU, Dropout(0.5)			
Linear 10, Softmax			

The architectures of the discriminator of pixel adaptation and reconstruction are the same as the encoder except that the output dimension of the last fully connected network is set as 1

Table 2 Unsupervised domain adaptation performance based on MNIST and USPS

	Source only	PixelDA	ADDA	CoGAN	UNIT	MCD	SBADA-GAN	CDRD	CYCADA	TPN	Ours
MNIST to USPS	0.822	0.959	0.894	0.957	0.959	0.965	**0.976**	0.951	0.956	0.921	**0.976**
USPS to MNIST	0.696	–	0.901	0.932	0.936	0.941	0.950	0.944	0.965	0.941	**0.972**

Bold values indicate the best performance
We report the best results obtained for the baseline methods from the published papers

distribution as

$$p_k = \begin{cases} \frac{1}{N_K}, & \{k|h_k > \tau\} \\ 0, & \{k|h_k \leq \tau\} \end{cases} \tag{10}$$

where N_K is the total number of category where $h_k > \tau$ (we empirically set $\tau = 0.03$ for the trade-off between category imbalance and wrong label prediction). We set the dimension of the encoder output as 10 which is the same as the number of categories from source domain, and experimentally set the dimension of local code as 512 initialized as $N \sim (0, 1)$. The details of the network are listed Table 1.

For the balanced setting, the results are shown in Table 2. It is clearly observed that our proposed method can outperform all other baselines under all scenarios by considering different protocols. For ADDA (Tzeng et al. 2017) and PixelDA (Bousmalis et al. 2017), only the feature representation induced from LeNet or reconstructed pixel was aligned with adversarial loss for domain adaptation, respectively, which may not be sufficient for domain adaptation task. Thus it is reasonable that ADDA and PixelDA can not achieve desired performance. Although MCD proposed to employ two discriminators in an elegant way, the semantic and pixel information between two domains were ignored. Though CBADA-GAN (Russo et al. 2018) can achieve good

Table 3 Imbalanced unsupervised domain adaptation performance based on MNIST and USPS

	Source only	ADDA	MCD	CDRD	CYCADA	ETN	Ours
MNIST to USPS (0–4)	0.916	0.919	0.887	0.941	**0.955**	0.860	0.951
USPS to MNIST (0–4)	0.778	0.764	0.902	0.934	0.954	0.857	**0.968**
MNIST to USPS (5–9)	0.618	0.656	0.669	0.776	0.898	0.660	**0.931**
USPS to MNIST (5–9)	0.597	0.782	0.829	0.831	0.901	0.798	**0.925**

Bold values indicate the best performance
(0–4) and (5–9) denotes only category 0–4 and 5–9 are contained in the target domain, respectively

performance when training on MNIST and testing on USPS, the performance regarding training on USPS and testing on MNIST cannot compete with CYCADA as only pixel adaptation is considered. For CoGAN (Liu and Tuzel 2016) and CDRD (Liu et al. 2018) baselines, the two distributions were aligned by a random generated vector with the adversarial loss. However, as have already been discussed in the existing works (e.g. Liu et al. 2017), simply generating images based on random noise may fail to capture generalized information, which limit their domain adaptation capability. UNIT (Liu et al. 2017) aimed to improve CoGAN by imposing image transfer regularization with reconstruction loss, which achieved better performance. CYCADA (Hoffman et al. 2018) considered both image transfer as well as feature level adaptation based on CycleGAN (Zhu et al. 2017). Last but not the least, though TPN (Pan et al. 2019) can induce theoretical lower bound of target domain risk, it may not be able to lead to an optimal embedding due to the network architecture. Our method can also be treated as aligning distribution based on pixel level and semantic level, as we impose a implicit distribution based on a Gaussian vectors shared by source and target domains with style transfer regularization. Moreover, we also take category information into account. We show in ablation study that these factors can jointly improve the performance of domain adaptation.

We then evaluate our method by considering imbalanced domain adaptation setting by only taking partial category information into account. The results are shown in Table 3. Based on the results, we observe that our proposed method can achieve significantly better performance compared with ADDA, MCD and CDRD. Though ADDA can perform better compared with directly training on source domain, the improvements are not significantly large, which shows that only applying semantic feature adaptation may not be sufficient. For MCD, though the adaptation is considered based on boundary level, it did not take imbalance setting into consideration, and both pixel and semantic feature adaptation were ignored, which also leads to poor performance under this setting. For CDRD, we observe that the performance is also not desired. We conjecture the reason that although disentanglement is considered, the categorical information was randomly generated and further imposed on conditional GAN, which may not be capable to handle the imbalanced setting. For CYCADA, it can achieve reasonably good performance under this scenario, which we conjecture that CycleGAN architecture can preserve category information during adaptation to some extent. We further consider to compare with the recently proposed Example Transfer Network (ETN) (Cao et al. 2019), which was designed for partial domain adaptation algorithm.[2] We find it cannot achieve desired performance in this scenario. We conjecture that it did not take autoencoder into considering thus the latent feature may not be able to preserve distinguishable information. Nevertheless, our algorithm can achieve the best performance in general, which indicate the importance of disentanglement through implicit autoencoders for unsupervised domain adaptation task.

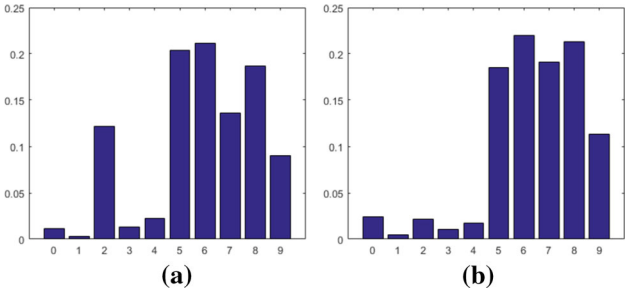

Fig. 4 Category output distribution on USPS dataset by considering adapting from MNIST to USPS (5–9). Left: category output obtained without semantic feature adaptation. Right: category output with semantic feature adaptation

As we adopt the semantic feature adaptation to obtain the Categorical prior for digit recognition task, we are also interested in how the semantic feature adaptation contributes in this imbalanced setting. To this end, we show the distribution of predicted output of target domain in Fig. 4 by considering the challenging case where MNIST is used as source domain and USPS (5–9) as target domain without semantic feature adaptation and with semantic feature adaptation during the first stage. As we can observe, a more reliable prior can be obtained by applying semantic feature adaptation which benefits domain adaptation under imbalanced setting. We also evaluate whether a reliable prior can be obtain through other adaptation, such as pixel adaptation based on our proposed framework, but found it may not achieve desired performance as semantic feature adaptation.

We further conduct experiments to understand the impact of different components of our proposed algorithm on the domain adaptation task under balanced category setting. Experimental results are shown in Table 4. "No Prior" means that we remove the adversarial loss of matching the output of encoder with the Category distribution. "No Pixel" means that we remove the pixel level adaptation. "No Recon" means that we remove reconstruction loss. "No Feat" means that we remove the feature adaptation. Moreover, we also consider using the deterministic reconstruction loss $L1$ and $L2$ as baseline instead of the adopted stochastic reconstruction loss (3).

From the table, we observe that removing the regularization based on the prior distribution, image translation and reconstruction, or replace the stochastic reconstruction loss with $L1$, $L2$ loss lead to poorer performance. This verifies

[2] We adopt the LeNet as backbone network, which is the benchmark for MNIST and USPS datasets.

Table 4 Impact of different components on performance

	MNIST to USPS	USPS to MNIST
No prior	0.739	0.755
No pixel	0.828	0.743
No recon	0.935	0.906
No feat	**0.976**	0.970
Ours (L1)	0.953	**0.972**
Ours (L2)	0.957	0.970
Ours (stochastic)	**0.976**	**0.972**

Bold values indicate the best performance

Table 5 Performance comparison of single-stage training and multi-stage training

	MNIST → USPS	USPS → MNIST
Multi-stage	0.976	0.972
Single-stage	0.977	0.974

the effectiveness of our proposed framework: (1) imposing pixel level adaptation can benefit domain adaptation task, which has also been discussed in existing works (Hoffman et al. 2018; Liu et al. 2017, 2018), (2) imposing prior constraint is significant to capture categorical information, as we observe it can lead to negative transfer without prior constraint. We conjecture the reason that the encoder can be overfitted to the classification loss thus the model fails to achieve disentanglement for target domain data, (3) imposing reconstruction loss can guarantee the latent code capture meaningful representation of the input images which also benefit domain adaptation, (4) we do not observe performance drop after removing semantic feature adaptation. However, the semantic feature adaptation term can help with training a more stable network especially for the imbalanced setting, (5) compared with using deterministic reconstruction loss $L1, L2$, imposing stochastic reconstruction can only capture the abstract and high-level information of given image (Makhzani 2018) which benefits disentangling feature representation learning and domain adaptation.

As we follow CYCADA (Hoffman et al. 2018) conducting a multi-stage training, it is valuable to investigate the performance by jointly training the whole objective in a single stage. The results are shown in Table 5. As we can observe, by jointly training the objective, we can even achieve slightly better performance, which is reasonable as training multiple loss jointly can help avoid the network overfit to one particular regularization. However, as also indicated in Hoffman et al. (2018), multi-stage training can benefit the situation where no sufficient GPU memory available and also help with preventing potential volatile training.

To further evaluate the effectiveness that our proposed framework which can learn disentangling representation, we use the one-hot vectors which represent digits from 0 to 9 together with randomly generated Gaussian vectors as the input of the generation model, which is trained by MNIST as the source domain and USPS as the target domain. The visualization results are shown in Fig. 5. As we can see, based on the cross-domain setting, the global code retains the label information of digit, while the local code captures the variations of in the style of the digits, which shows the effectiveness of our algorithm for disentangling representation learning and can capture the transferable information between source and target domains, which benefit unsupervised domain adaptation.

5.1.3 SVHN ↔ MNIST

Subsequently, we consider domain adaptation between SVHN and MNIST which have distinct properties. SVHN contains images with diverse color background with multiple digits which are very blurred while MNIST is in grayscale and much sharper. In order to conduct domain adaptation between SVHN and MNIST, we first convert the images in MNIST to RGB and then resize the images to 32×32, as suggested in Liu et al. (2017), and further conduct processing (French et al. 2017) on MNIST when using it as source domain. We adopt the model in Saito et al. (2018) as the encoder. We set the batch size to 128, and follow the protocol in Bousmalis et al. (2017) and Russo et al. (2018) for parameter setting, which leads to $\lambda_0 = 10, \lambda_1 = 2, \lambda_2 = \lambda_3 = 1$ and $\lambda_4 = 0.01$. We use Adam (Kingma and Ba 2014) to optimize the model with learning rate as 2×10^{-4}, which is also consistent with (Makhzani and Frey 2017). We follow MNIST ↔ USPS setting for prior generation. The dimensions of the encoder output and local code are set as 10 and 512, respectively. The local code is initialized as $N \sim (0, 1)$. The details of the network are listed in Table 6.

As can be observed from Table 7, all domain adaptation algorithms can outperform the baseline by training only based on source domain. The performance improvement of ADDA is much smaller compared with other methods, which is consistent with the results in Table 2. The performance of SBADA-GAN is also not desired, as feature level adaptation is missing. However, an interesting observation is that, MCD can achieve much better performance compared with UNIT and CYCADA. We conjecture two possible reasons, (1) only conducting image style translation is not sufficient to handle the domain shift when source and target domain are distinct, as it only considers distribution divergence based on data level, (2) aligning classifier is also important for domain adaptation task, as it can be treated as reducing distribution divergence by considering category distribution divergence based on conditional distribution with given input data. Our

Fig. 5 Image generation of MNIST and USPS with one-hot vector as label (each row represents a label) and random Gaussian noise as style. Left: MNIST, right: USPS

Table 6 Network architectures used for SVHN ↔ MNIST

Encoder	Generator	Disc. prior	Disc. feat
Conv5, maps 64, BN 64, ReLU	Linear 4096, BN 4096, ReLU	Linear 100, ReLU	Linear 2048, ReLU
Max Pool 3, Stride 2	DeConv4, Stride 2, maps 128, BN 128, ReLU	Linear 100, ReLU	Linear 2048, ReLU
Conv5, maps 128, BN 128, ReLU	DeConv4, Stride 2, maps 64, BN 64, ReLU	Linear 1	Linear 1
Max Pool 3, Stride 2	DeConv4, Stride 2, maps 6, tanh		
Linear 3072, BN 3072, ReLU, Dropout(0.5)			
Linear 2048, BN 2048, ReLU			
Linear 10, Softmax			

The architectures of discriminator of pixel adaptation and reconstruction are the same as the encoder except that the output dimension of the last fully connected network is set as 1

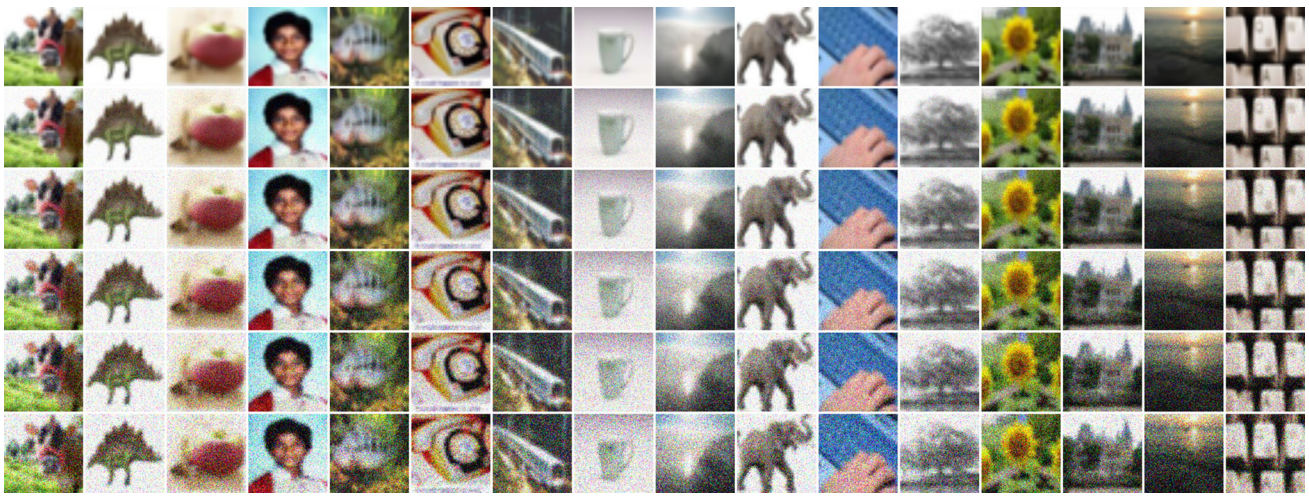

Fig. 6 Examples of CIFAR-100 datasets. The first row shows the clean samples without Gaussian noise distortion. The second to the last rows show the images with five different severity levels ranging from 1 to 5

Table 7 Unsupervised domain adaptation on SVHN and MNIST

	SVHN to MNIST	MNIST to SVHN
Source only	0.671	0.260
ADDA	0.760	–
UNIT	0.905	–
MCD	0.962	–
CYCADA	0.904	–
SBADA-GAN	0.761	0.611
Ours	**0.978**	**0.662**

Bold values indicate the best performance

algorithm achieves the best performance, as we consider both pixel- and semantic-level adaptation, as well as category level regularization.

5.2 Cross-Domain Object Recognition

Next, we consider cross-domain object recognition task based on CIFAR-100 dataset (Krizhevsky and Hinton 2009). The CIFAR-100 dataset consists of 60,000 32×32 color images in 100 classes. In particular, we consider the clean CIFAR-100 as source domain and the corrupted CIFAR-100 by adding Gaussian noise with five different severity levels ranging from 1 to 5 ($std = \{0.04, 0.08, 0.12, 0.15, 0.18\}$ by normalizing the clean images in the range of [0, 1]) as target domain. Such setting is one of recent benchmarks to evaluate the robustness of deep neural networks (Hendrycks and Dietterich 2019). We show several examples of clean and corrupted image from CIFAR-100 dataset in Fig. 6. We also utilize wider ResNet (Zagoruyko and Komodakis 2016) with depth as 40 and widen factor as 2 as the encoder. We adopt the algorithms including ADDA and MCD, where the fully connected layer is adopted as classifier and the remaining as feature extractor, as baselines for comparison. For the discriminator of ADDA, we use a similar architecture as adopted in Tzeng et al. (2017). Regarding the architecture, parameter setting and prior generation, we follow the same setting for SVHN/MNIST task except that we set the probability p_k of Categorical distribution as $\frac{1}{100}$ with the number of category as 100 and adopt wider ResNet as encoder with the dimension of output as 100. For other methods, we report their best results on the test data by varying their parameters in a wide range. The results are shown in Table 8. As we can see, even adding subtle Gaussian noise on target domain can lead to huge performance drop (the recognition drops from 0.765 to 0.423 when setting the standard value of Gaussian noise to 0.04). Nevertheless, our proposed method can also handle the difficult domain adaptation task with large number of categories. We also observe that our proposed method can achieve better performance compared with ADDA and MCD, which is reasonable as we include prior regularization as well as pixel adaptation in our proposed method. Compared with ADDA, MCD achieves slightly worse performance. We conjecture the reasons that directly conducting domain alignment based on category level may not be suitable to handle the scenario where there are large number of categories involves. This observation further justifies our motivation to conduct feature disentangling. Noted that we did not compare with other autoencoder based methods, as it is not clear how to build a suitable encoding and decoding architecture based on wider ResNet while we can directly adopting wider ResNet as the encoder.

Table 8 Unsupervised domain adaptation performance based on CIFAR-100 and corrupted CIFAR-100

Target severity	1	2	3	4	5	Clean
Source only	0.423	0.255	0.155	0.121	0.104	0.765
ADDA	0.431	0.283	0.191	0.145	0.129	–
MCD	0.429	0.260	0.173	0.132	0.107	–
Ours	**0.469**	**0.328**	**0.233**	**0.174**	**0.151**	–

Bold values indicate the best performance
We conduct the experiments for ten times and report the average performance

Fig. 7 Examples of GTA5 (first row), SYNTHIA (second row) and CityScape datasets (third row)

5.3 Cross-Domain Semantic Segmentation

In this section, we evaluate our proposed method on cross-domain semantic image segmentation task between the synthetic dataset GTA5 (Richter et al. 2016), SYNTHIARAND-CITYSCAPES (SYNTHIA) (Ros et al. 2016) and real-world dataset CITYSCAPES (Cordts et al. 2016), where one of the synthetic dataset is used as source domain and CITYSCAPES for target domain. We show several examples from these three datasets in Fig. 7.

Table 9 Adaptation results based on mIoU (%) from GTA5 to CityScape based on FCN8s

	Road	Sidewalk	Building	Wall	Fence	Pole	Traffic light	Traffic sign	Vegetation	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Motorbike	Bicycle	mIoU
Source only	26.0	14.9	65.1	5.5	12.9	8.9	6.0	2.5	70.0	2.9	47.0	24.5	0.0	40.0	12.1	1.5	0.0	0.0	0.0	17.9
FCN Wld (Hoffman et al. 2016)	70.4	32.4	62.1	14.9	5.4	10.9	14.2	2.7	79.2	21.3	64.6	44.1	4.2	70.4	8.0	7.3	0.0	3.5	0.0	27.1
CYCADA (feature) (Hoffman et al. 2018)	85.6	30.7	74.7	14.4	13.0	17.6	13.7	5.8	74.6	15.8	69.6	38.2	3.5	72.3	16.0	5.0	0.1	3.6	0.0	29.2
CYCADA (pixel) (Hoffman et al. 2018)	83.5	**38.3**	76.4	20.6	16.5	22.2	26.2	21.9	80.4	28.7	65.7	49.4	4.2	74.6	16.0	26.6	2.0	8.0	0.0	34.8
CYCADA (feat + pixel) (Hoffman et al. 2018)	85.2	37.2	76.5	21.8	15.0	**23.8**	22.9	21.5	80.5	31.3	60.7	**50.5**	**9.0**	76.9	17.1	28.2	4.5	9.8	0.0	35.4
MCD (Saito et al. 2018)	86.4	8.5	76.1	18.6	9.7	14.9	7.8	0.6	82.8	**32.7**	**71.4**	25.2	1.1	76.3	16.1	17.1	1.4	0.2	0.0	28.8
AdaptSeg (Tsai et al. 2018)	87.3	29.8	78.6	21.1	18.2	22.5	21.5	11.0	79.7	29.6	71.3	46.8	6.5	80.1	**23.0**	26.9	0.0	10.6	0.3	35.0
AdvEnt (Vu et al. 2019)	86.9	28.7	**78.7**	**28.5**	**25.2**	17.1	20.3	10.9	80.0	26.4	70.2	47.1	8.4	**81.5**	26.0	17.2	**18.9**	**11.7**	1.6	36.1
Ours	**88.2**	37.8	76.5	23.0	18.4	23.0	**27.7**	**22.1**	**83.1**	29.6	69.4	48.2	4.0	73.9	16.7	**28.4**	2.3	10.7	**5.6**	**36.3**

Bold values indicate the best performance
We report the best results obtained for the baseline methods from the published papers

Table 10 Adaptation results based on mIoU (%) from SYNTHIARAND-CITYSCAPE to CityScape based on FCN8s

	Road	Sidewalk	Building	Wall	Fence	Pole	Traffic light	Traffic sign	Vegetation	Sky	Person	Rider	Car	Bus	Motorbike	Bicycle	mIoU
Source only	5.6	11.2	59.6	0.8	**0.5**	21.5	**8.0**	5.3	72.4	75.6	35.1	**9.0**	23.6	4.5	0.5	**18.0**	22.0
FCN Wld (Hoffman et al. 2016)	11.5	19.6	30.8	4.4	0.0	20.3	0.1	11.7	42.3	68.7	**51.2**	3.8	54.0	3.2	0.2	0.6	20.2
AdaptSeg (Tsai et al. 2018)	**78.9**	29.2	75.5	–	–	–	0.1	4.8	**72.6**	**76.7**	43.4	8.8	71.1	16.0	**3.6**	8.4	–
AdvEnt (Vu et al. 2019)	67.9	29.4	71.9	6.3	0.3	19.9	0.6	2.6	74.9	74.9	35.4	**9.6**	67.8	**21.4**	4.1	15.5	31.4
Ours	77.3	**32.0**	**76.9**	**9.1**	**0.5**	**23.3**	4.4	**12.9**	70.0	71.8	50.6	8.8	**73.9**	20.0	3.4	10.1	**34.0**

Bold values indicate the best performance
We report the best results obtained for the baseline methods from the published papers

The GTA5 dataset consists of 24,966 images with the resolution of 1914 × 1052 by extracting frames from the computer game Grand Theft Auto. The SYNTHIA dataset contains 9400 synthesized images taken from a virtual array of cameras. The CITYSCAPES dataset contains real urban street images with 5000 images which as splitted into three subsets, 2975 images in train set, 500 images in the val set and 1595 images in the test set. We follow our baseline methods by considering the val set in CITYSCAPES as our target domain. We use 19 common classes between GTA5 and CITYSCAPES, and 16 common classes between SYNTHIA and CITYSCAPES as our label. We report the final performance based on mean intersection-over-union (mIoU). We compare with the recent unsupervised domain adaptation methods, including FCN in the wild (Hoffman et al. 2016), MCD (Saito et al. 2018), AdaptSeg (Tsai et al. 2018), CYCADA (Hoffman et al. 2018) and the recent proposed AdvEnt (Vu et al. 2019).

Implementation In all our experiments, the images are resized to 1024 × 512. For efficiency, we use the one-hot resized groundtruth annotation $\mathbf{y}_S \in \mathbf{Y}_S$ of the input image from source domain as the categorical prior. We experimentally set $\lambda_0 = 1$, $\lambda_1 = \lambda_2 = 0.001$ and $\lambda_3 = \lambda_4 = 1e^{-5}$. In order to fairly compare with other baseline algorithms, we only consider FCN8s (Long et al. 2015) as the encoder with batch size as 1. The FCN8s is initialized with the weights of the VGG16 (Simonyan and Zisserman 2014) model trained on Imagenet (Deng et al. 2009). For the local code, we generate random Gaussian distribution $N \sim (0, 1)$ with the same size of the encoder output. Regarding the generator, we follow the architecture in Zhu et al. (2017) with 6 residue blocks (the number of feature map is set to $2 \times \#Cat$ for input and 6 for output), where $\#Cat$ is the number of category. Regarding the discriminator, we impose five convolutional layers with kernel size 4×4 with stride of 2. The number of channels in each convolutional layer are set as 64, 128, 256, 512, 1. Similar to Hoffman et al. (2016), we consider the output of the last layer before pixel prediction for semantic feature adaptation. For evaluation, we first obtain the predictions on the resized image and then upsample the prediction to get 2048 × 1024 to get the final label map.

We first summarize the results in Table 9 by considering GTA5 as source domain. We observe that domain adaptation benefits cross-domain semantic segmentation task. FCN in the wild and MCD achieve relatively poorer performance as they only consider either semantic level adaptation or classifier level adaptation, respectively. CYCADA achieves better performance as it take pixel and semantic information into consideration. Similar to CYCADA, we also conduct domain adaptation in multiple levels. However, as we also consider Categorical prior matching based on the output of the encoder from the target domain, which can be treated as category regularization, it is reasonable that our method outperforms CYCADA. Although our method only achieves slightly better performance compared with the AdvEnt, our method is more general as AdvEnt is purely designed for segmentation task.

Noted that the segmentation performance for some categories (e.g. bicycle) are not desired. We conjecture the reason that it can be related the imbalance segmentation labelling which our prior relies on. As the instance segmentation can associate to the proportion of each instance as well as its layout. Thus, the pixels are likely to be categorized to the instances which appear more frequently than others.

We then analyze the performance in Table 10 by considering SYNTHIARAND-CITYSCAPE as source domain. Generally, the cross-domain performance is worse compared with the results by using GTA5 as source domain. We conjecture the reason that scene images in SYNTHIARAND-CITYSCAPE contains more diverse viewpoints than the ones in GTA5. Nevertheless, we can still achieve better performance compared with other baselines. Noted that in AdaptSeg (Tsai et al. 2018), only 13 classes are adopted for evaluation. We can still outperform AdaptSeg as we can achieve 39.3% mIoU while AdaptSeg achieved 37.6%.

Finally, we show some visualization results in Fig. 8 by considering GTA5 as source domain. We observe that, compared with the results without adaptation, we can largely improve the segmentation results. However, we observe that our results are to some extent over-smoothed, which may due to the reason that we resize the input image from 2048 × 1024 to 1024 × 512 and upsample again, which make the final segmentation map over-smooth. Such phenomenon has also been observed in traditional segmentation task (e.g. RefineNet Lin et al. 2017) based on intra-domain setting. To deal with this problem, one may further consider a Condition-Random-Field process to conduct post-processing of the output to obtain better performance.

Similar to digit recognition task, we also consider to conduct ablation study by analyzing the impact of different components for cross-domain segmentation task. In particular, we consider GTA5 as source domain and Cityscape as target domain. The results are shown in Table 11. As we can see, compared with semantic feature adaptation, the prior regularization, pixel adaptation and reconstruction regularization play a more important role for cross-domain segmentation task. Such results are also consistent with the finding in Hoffman et al. (2016, 2018). Nevertheless, we find adopting semantic feature adaptation can lead to slightly better performance. We conjecture the reason that a more reliable latent code from target domain can be obtained, which can further benefit prior alignment in the second stage. On the other hand, we also find introducing prior adaptation can lead to significant improvement of final performance, which is reasonable as it can benefit feature disentangling of the target domain.

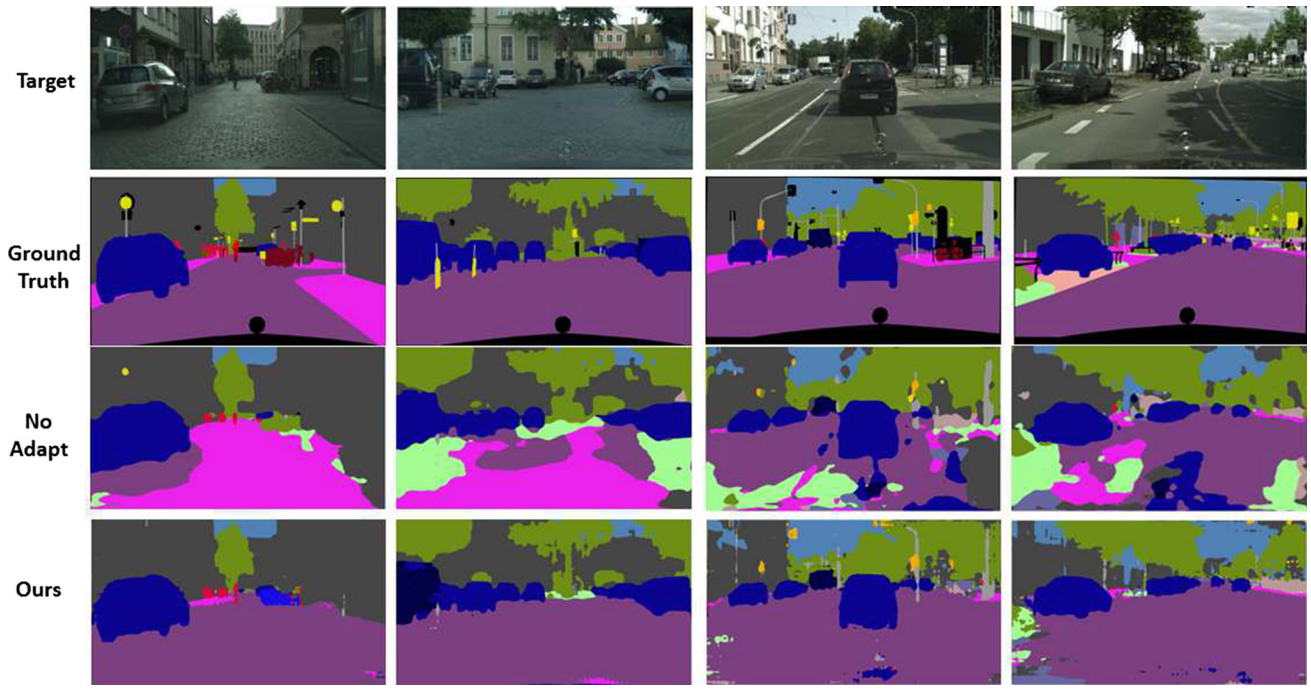

Fig. 8 Example results of our proposed method from GTA5 to CITYSCAPES

5.4 Discussion of Categorical Prior

We observe that the Categorical prior plays an important role in both recognition task and segmentation task. The idea of categorical prior by generating one-hot vector was introduced in Makhzani et al. (2015), which is for digit generation and semi-supervised learning task. We adopt such idea and further introduce a threshold for our digit recognition domain adaptation task. We have tried to consider the category probability for each digit of source domain to construct the prior, and find there is almost no difference compared with the prior by generating using the equally distributed Categorical distribution.

We also used the randomly generated one-hot vector for each resized pixel for cross-domain segmentation task as well. However, compared with digit recognition task, segmentation task is more difficult due to its imbalance category as well as the category layout. We also consider to adopt the category probability from source domain to construct the prior by assuming that the category proportions are consistent between source and target domain, but find the performance is not desired. We conjecture that the layout is also important to construct the prior for segmentation task. As both category proportion and layout can be important, we directly impose the groundtruth label from source domain, which can also be treated as Catagorical distributed, as the prior. How to propose a reasonable prior for cross-domain segmentation task will be investigated in our future work.

Table 11 Impact of different components on performance

	GTA5 to CityScape
No prior	24.4
No pixel	32.5
No recon	33.2
No feat	36.1
Ours	**36.3**

Bold value indicates the best performance

6 Conclusion

In this paper, we present a deep learning framework based on implicit autoencoders for unsupervised domain adaptation task in the wild. The main idea of our proposed framework is to conduct disentanglement based on latent feature representation. We show that, the global code which captures the categorical information can be learned with the regularization of prior distribution matching while the style information can be captured by implicit conditional likelihood distribution, which make our proposed unsupervised domain adaptation framework effective. We conduct experiments based on digit recognition, object recognition and semantic segmentation. The experimental results indicate that our proposed framework is effective to handle unsupervised domain adaptation task in the wild.

Acknowledgements The research work was done at the Rapid-Rich Object Search (ROSE) Lab, Nanyang Technological University. This research is supported in part by the Wallenberg-NTU Presidential Postdoctoral Fellowship, the NTU-PKU Joint Research Institute, a collaboration between the Nanyang Technological University and Peking University that is sponsored by a donation from the Ng Teng Fong Charitable Foundation, and the Science and Technology Foundation of Guangzhou Huangpu Development District under Grant 201902010028.

References

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., & Vaughan, J. W. (2010). A theory of learning from different domains. *Machine Learning*, *79*(1–2), 151–175.

Benford, F. (1938). The law of anomalous numbers. *Proceedings of the American Philosophical Society*, *78*(4), 551–572.

Bousmalis, K., Silberman, N., Dohan, D., Erhan, D., & Krishnan, D. (2017). Unsupervised pixel-level domain adaptation with generative adversarial networks. In *CVPR*.

Cao, Z., You, K., Long, M., Wang, J., & Yang, Q. (2019). Learning to transfer examples for partial domain adaptation. In *CVPR*.

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., & Abbeel, P. (2016). Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *NeurIPS*.

Chen, Y., Li, W., & Van Gool, L. (2018). Road: Reality oriented adaptation for semantic segmentation of urban scenes. In *CVPR*.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., et al. (2016). The cityscapes dataset for semantic urban scene understanding. In *CVPR*.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *CVPR*.

French, G., Mackiewicz, M., & Fisher, M. (2017). *Self-ensembling for visual domain adaptation*. arXiv preprint arXiv:1706.05208.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., et al. (2016). Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, *17*, 2096–2030.

Ghifary, M., Kleijn, W. B., Zhang, M., Balduzzi, D., & Li, W. (2016). Deep reconstruction-classification networks for unsupervised domain adaptation. In *ECCV*.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., et al. (2014). Generative adversarial nets. In *NeurIPS*.

Haeusser, P., Frerix, T., Mordvintsev, A., & Cremers, D. (2017). Associative domain adaptation. In *CVPR*.

Hendrycks, Da., & Dietterich, T. (2019). Benchmarking neural network robustness to common corruptions and perturbations. In *ICLR*.

Hoffman, J., Tzeng, E., Park, T., Zhu, J.-Y., Isola, P., Saenko, K., et al. (2018). Cycada: Cycle-consistent adversarial domain adaptation. In *ICML*.

Hoffman, J., Wang, D., Yu, F., & Darrell, T. (2016). *Fcns in the wild: Pixel-level adversarial and constraint-based adaptation*. arXiv preprint arXiv:1612.02649.

Huang, J., Gretton, A., Borgwardt, K. M., Schölkopf, B., & Smola, A. J. (2006). Correcting sample selection bias by unlabeled data. In *NeurIPS*.

Huang, X., Liu, M.-Y., Belongie, S., & Kautz, J. (2018). *Multimodal unsupervised image-to-image translation*. arXiv preprint arXiv:1804.04732.

Hull, J. J. (1994). A database for handwritten text recognition research. In *PAMI*.

Isola, P., Zhu, J.-Y., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. arXiv preprint.

Kingma, D. P., & Ba, J. (2014). *Adam: A method for stochastic optimization*. arXiv preprint arXiv:1412.6980.

Kingma, D. P., & Welling, M. (2013). *Auto-encoding variational bayes*. arXiv preprint arXiv:1312.6114.

Krizhevsky, A., & Hinton, G. (2009). Learning multiple layers of features from tiny images.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *NeurIPS*.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278–2324.

Levinkov, E., & Fritz, M. (2013). Sequential Bayesian model update under structured scene prior for semantic road scenes labeling. In *CVPR*.

Lin, G., Milan, A., Shen, C., & Reid, I. (2017). Proceedings of the IEEE conference on computer vision and pattern recognition.

Liu, M.-Y., Breuel, T., & Kautz, J. (2017). Unsupervised image-to-image translation networks. In *NeurIPS*.

Liu, M.-Y., & Tuzel, O. (2016). Coupled generative adversarial networks. In *NeurIPS*.

Liu, Y.-C., Yeh, Y.-Y., Fu, T.-C., Wang, S.-D., Chiu, W.-C., & Wang, Y.-C. F. (2018). Detach and adapt: Learning cross-domain disentangled deep representation. In *CVPR*.

Long, M., Cao, Y., Wang, J., & Jordan, M. (2015). Learning transferable features with deep adaptation networks. In *ICML*.

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *CVPR*.

Long, M., Zhu, H., Wang, J., & Jordan, M. I. (2016). Unsupervised domain adaptation with residual transfer networks. In *NeurIPS*.

Makhzani, A. (2018). *Implicit autoencoders*. arXiv preprint arXiv:1805.09804.

Makhzani, A., & Frey, B. J. (2017). Pixelgan autoencoders. In *NeurIPS*.

Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., & Frey, B. (2015). *Adversarial autoencoders*. arXiv preprint arXiv:1511.05644.

Ming Harry Hsu, T., Yu Chen, W., Hou, C.-A., Hubert Tsai, Y.-H., Yeh, Y.-R., & Frank Wang, Y.-C. (2015). Unsupervised domain adaptation with imbalanced cross-domain data. In *ICCV*.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., & Ng, A. Y. (2011). Reading digits in natural images with unsupervised feature learning. In *NeurIPS Workshop*, volume 2011.

Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., & Ng, A. Y. (2011). Multimodal deep learning. In *ICML*.

Pan, S. J., Tsang, I. W., Kwok, J. T., & Yang, Q. (2011). Domain adaptation via transfer component analysis. In *TNN*.

Pan, S. J., & Yang, Q. (2010). A survey on transfer learning. In *TKDE*.

Pan, Y., Yao, T., Li, Y., Wang, Y., Ngo, C.-W., & Mei, T. (2019). Transferrable prototypical networks for unsupervised domain adaptation. In *CVPR*.

Richter, S. R., Vineet, V., Roth, S., & Koltun, V. (2016). Playing for data: Ground truth from computer games. In *ECCV*.

Ros, G., Sellart, L., Materzynska, J., Vazquez, D., & Lopez, A. M. (2016). The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3234–3243).

Russo, P., Carlucci, F. M., Tommasi, T., & Caputo, B. (2018). From source to target and back: Symmetric bi-directional adaptive gan. In *CVPR*.

Saito, K., Watanabe, K., Ushiku, Y., & Harada, T. (2018). Maximum classifier discrepancy for unsupervised domain adaptation. In *CVPR*.

Sankaranarayanan, S., Balaji, Y., Jain, A., Lim, S. N., & Chellappa, R. (2017). *Unsupervised domain adaptation for semantic segmentation with gans*. arXiv preprint arXiv:1711.06969.

Schölkopf, B., Janzing, D., Peters, J., Sgouritsa, E., Zhang, K., & Mooij, J. (2012). On causal and anticausal learning. In *ICML*.

Shrivastava, A., Pfister, T., Tuzel, O., Susskind, J., Wang, W., & Webb, R. (2017). Learning from simulated and unsupervised images through adversarial training. In *CVPR*, Vol. 2, p. 5.

Simonyan, K., & Zisserman, A. (2014). *Very deep convolutional networks for large-scale image recognition*. arXiv preprint arXiv:1409.1556.

Taigman, Y., Polyak, A., & Wolf, L. (2016). *Unsupervised cross-domain image generation*. arXiv preprint arXiv:1611.02200.

Tsai, Y.-H., Hung, W.-C., Schulter, S., Sohn, K., Yang, M.-H., & Chandraker, M. (2018). *Learning to adapt structured output space for semantic segmentation*. arXiv preprint arXiv:1802.10349.

Tzeng, E., Hoffman, J., Darrell, T., & Saenko, K. (2015). Simultaneous deep transfer across domains and tasks. In *ICCV*.

Tzeng, E., Hoffman, J., Saenko, K., & Darrell, T. (2017). Adversarial discriminative domain adaptation. In *CVPR*.

Vu, T.-H., Jain, H., Bucher, M., Cord, M., & Pérez, P. (2019). Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *CVPR*.

Wang, T.-C., Liu, M.-Y., Zhu, J.-Y., Tao, A., Kautz, J., & Catanzaro, B. (2017). *High-resolution image synthesis and semantic manipulation with conditional gans*. arXiv preprint arXiv:1711.11585.

Zagoruyko, S., & Komodakis, N. (2016). *Wide residual networks*. arXiv preprint arXiv:1605.07146.

Zhang, Y., David, P., & Gong, B. (2017). Curriculum domain adaptation for semantic segmentation of urban scenes. In *ICCV*.

Zhang, Y., Qiu, Z., Yao, T., Liu, D., & Mei, T. (2018). Fully convolutional adaptation networks for semantic segmentation. In *CVPR*.

Zhu, J.-Y., Park, T., Isola, P., & Efros, A. A. (2017). *Unpaired image-to-image translation using cycle-consistent adversarial networks*. arXiv preprint arXiv:1703.10593.