JOINT GLOBAL-LOCAL ALIGNMENT FOR DOMAIN ADAPTIVE SEMANTIC SEGMENTATION

Sudhir Yarram[†] *Ming Yang*[‡] *Junsong Yuan*[†] *Chunming Qiao*[†]

[†] University at Buffalo [‡]Horizon Robotics

ABSTRACT

Unsupervised domain adaptation has shown promising results in leveraging synthetic (source) images for semantic segmentation of real (target) images. One key issue is how to align data distributions between the source and target domains. Adversarial learning has been applied to align these distributions. However, most existing approaches focus on aligning the output distributions related to image (global) segmentation. Such global alignment may not result in effective alignment due to the inherent high dimensionality feature space involved in the alignment. Moreover, global alignment might be hindered by the noisy outputs corresponding to background pixels in the source domain. To address this limitation, we propose a local output alignment. Such an approach can also mitigate the influences of noisy background pixels from the source domain when performing the local alignment. Our experiments show that by adding local output alignment into various global alignment based domain adaptation, our joint global-local alignment methods improves semantic segmentation. Code is available at https://github.com/skrya/globallocal.

Index Terms— semantic segmentation, domain adaptation, global-local alignment.

1. INTRODUCTION

Semantic segmentation aims to assign dense labels to all the pixels in an image. Training segmentation models generally needs a lot of expensive labeled data. To alleviate this, recent approaches leverage synthetic image data to train image segmentation. Despite the effectiveness in training, unfortunately, the models often do not generalize well to target (real) scenes, especially when there is a domain gap that might arise due to different lighting conditions and weather conditions.

Unsupervised domain adaptation (UDA) [1, 2] has been proposed to close the domain gap between a source (synthetic) domain and a target (real) domain to improve semantic segmentation. A prominent approach is global (image-level) output distribution alignment [3] using adversarial learning [4, 2, 5], which is a min-max game played between a segmentation model and a discriminator. While the segmentation model produces outputs for semantic segmentation (*i.e.,* $\boldsymbol{p}_{x_s}$ for source sample, $\boldsymbol{p}_{x_t}$ for target sample, where $\boldsymbol{p}_{x_s}, \boldsymbol{p}_{x_t} \in \mathbb{R}^{H \times W \times K}$, where H, W are height and width of the image and K is the number of categories), the discriminator distinguishes whether those global ($\mathbb{R}^{H \times W \times K}$) outputs are from the source or target image.

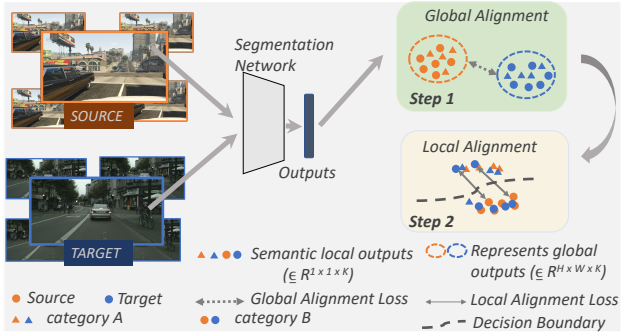

Fig. 1. **Illustration of joint global-local alignment as a two-step approach.** For the purpose of illustration, we only show adaptation across an image taken from both domains. Global alignment involves inherent high dimensionality in image (global) segmentation output (*i.e.,* $\mathbb{R}^{H \times W \times K}$) alignment. Here H, W are height and width of the image and K is the number of categories. Such an alignment might not conduct good alignment of local outputs (*i.e.,* $H \times W$ $\mathbb{R}^{1 \times 1 \times K}$ local output). So, we propose a two-step alignment, where we conduct the local alignment on top of a global alignment.

However, such a UDA approach has two major limitations. First, even though aligning the global output distributions results in similar overall distributions for the two domains, it does not ensure alignment of various local outputs (*i.e.,* $H \times W$ $\mathbb{R}^{1 \times 1 \times K}$ outputs that constitute $\boldsymbol{p}_{x_s}$ or $\boldsymbol{p}_{x_t}$). This is due to the inherent high dimensionality of $\boldsymbol{p}_{x_s}, \boldsymbol{p}_{x_t}$ that are involved in such a global alignment. Also, as these local outputs encode semantics relevant to diverse categories of objects, they are crucial for better segmentation performance. Conducting local alignment can enhance the semantic cues present in these local outputs to be more relevant to a particular category. This, in turn helps segmentation model to produce better segmentation for the target domain.

Second, various source datasets [6, 7] contain background pixels in the training dataset which are not used for training the segmentation model. For example, synthetic dataset, GTA5 contains 11% of background pixels and the outputs corresponding to these background pixels are still part of global output that is aligned across domains. While these outputs do not encode relevant information to help do-

main alignment, they act more as noise and hinder alignment of global distributions across domains.

To tackle these issues, we propose a local output alignment across source and target domains, on top of a global alignment, hence referred as global-local alignment (see Fig. 1). To achieve local alignment, we utilize pixel-level discriminator proposed by [8]. Meanwhile, as we tackle local outputs during adversarial training, we can easily discard the outputs corresponding to the noisy source background pixels. We leverage the source ground truth to achieve this. In experiments, we perform training on synthetic images (GTA5, SYNTHIA) and testing on real images (Cityscapes) to demonstrate the effectiveness of our approach.

2. RELATED WORK

Unsupervised domain adaptation (UDA) aims to rectify the domain mismatch and adapt the models towards better performance on real-world data. Several domain adaptation methods [9, 10, 3, 11, 12, 13, 14] for semantic segmentation have been proposed. Inspired by Generative adversarial Networks [4], methods [15, 5] use adversarial learning. Adversarial learning based methods can be mainly divided into image-translation and alignment based.

Inspired by the recent advances in image synthesis, CyCADA [16] proposed to generate target images conditioned on the source, which can help reduce the domain discrepancy before training segmentation models. In addition, CyCADA also uses adversarial loss to perform feature-level alignment. Moreover, they use self-training to boost the results further.

Image translation based methods can be considered to be image-level alignment based methods. At the output-level, [3] proposes AdapSegNet, in which they address domain discrepancy by structural output alignment. CLAN [17] leverages the attention mechanism and the class aware adversarial loss to further improve the performance. More recently, [11] performs output space alignment based on the entropy maps of pixel-wise predictions. At feature-level, [18] explores fine-grained feature alignment while preserving the internal structure of semantics across domains. Different from above methods that mainly conduct domain adaptation in a single step, we take a two-step approach to conduct domain adaptation. We conduct global alignment followed by local alignment. Moreover, the above methods mainly focus on global alignment while we focus on both global and local alignment. Also, none of the existing UDA methods focus on discarding the noisy features/outputs belonging to the background pixels. In contrast, through local alignment we can discard the features/outputs belonging to noisy background pixels from the source domain.

3. PROPOSED APPROACH

In this section, we describe our adversarial learning framework (see Fig. 2). We first introduce how to conduct global alignment, then the local alignment. We also present mixture module, which is used to enhance the local alignment.

We focus on unsupervised domain adaptation for semantic segmentation, a pixel-level classification task. It deals with source images $\boldsymbol{X}_S \in \mathbb{R}^{H\times W\times 3}$ with corresponding K-category segmentation ground truth denoted by $\boldsymbol{Y}_S \in \{1,\dots,K\}^{H\times W}$, where H and W are the height and width of the images. Similarly, the target images are denoted by $\boldsymbol{X}_T \in \mathbb{R}^{H\times W\times 3}$ *without ground truth*. The goal is to learn a model that can effectively perform semantic segmentation on the target dataset, over a set of predefined K categories. Given a source image $\boldsymbol{x}_s \in \mathbb{R}^{H\times W\times 3}$ with associated segmentation ground truth $\boldsymbol{y}_{x_s}$, the segmentation model $\boldsymbol{G}$ takes $\boldsymbol{x}_s$ as input and generates output $\boldsymbol{p}_{x_s} = \boldsymbol{G}(\boldsymbol{x}_s)$, $\boldsymbol{G}$: $\mathbb{R}^{H\times W\times 3} \mapsto \mathbb{R}^{H\times W\times K}$. The segmentation loss $\mathcal{L}_{seg}^{src}$ is a cross-entropy loss.

Global Alignment. To tackle the domain issue, global alignment through adversarial learning is proposed. It is a min-max game played between segmentation network $\boldsymbol{G}$ and a discriminator network $\boldsymbol{D}$. With this, for an input $\boldsymbol{x}_s$, $\boldsymbol{x}_t$, $\boldsymbol{G}$ outputs $\boldsymbol{p}_{x_s}$, $\boldsymbol{p}_{x_t}$ respectively. While $\boldsymbol{D}$ tries to discriminate if output $\boldsymbol{p}_{x_s}$, $\boldsymbol{p}_{x_t}$ is from source or target domain, $\boldsymbol{G}$ tries to confuse the discriminator. The objective can be formulated as :

$$\mathcal{L}_{global} = \frac{1}{|\boldsymbol{X}_S|}\sum_{\boldsymbol{x}_s\in\boldsymbol{X}_S}[\log \boldsymbol{D}(\boldsymbol{G}(\boldsymbol{x}_s))] + \frac{1}{|\boldsymbol{X}_T|}\sum_{\boldsymbol{x}_t\in\boldsymbol{X}_T}[1-\log(\boldsymbol{D}(\boldsymbol{G}(\boldsymbol{x}_t)))]. \tag{1}$$

The above objective seeks to align the predicted output distribution of the target data with predicted source output distribution. However, these methods adapt outputs at a global level ($\mathbb{R}^{H\times W\times K}$). In other words, it adapts all $H\times W$ local ($\mathbb{R}^{1\times 1\times K}$) output combined. In this work, we call a output of $\boldsymbol{p}_{x_s}$/$\boldsymbol{p}_{x_t}$ at a spatial location (i,j), *i.e.,* $\boldsymbol{p}_{x_s}^{(i,j)}$/$\boldsymbol{p}_{x_t}^{(i,j)}$ $\in \mathbb{R}^{1\times 1\times K}$ a local output. In the following, we introduce the local alignment, and then mixture module to conduct such an alignment.

Local Alignment. Inspired by [8], we utilize the pixel-level discriminator as local discriminator, dedicated to adaptation of local outputs across domains. Note that we perform local alignment for the outputs that are already globally aligned through global alignment. The local discriminator takes output at a spatial location (i,j) (*i.e.,* local output) from either source ($\boldsymbol{p}_{x_s}$) or target ($\boldsymbol{p}_{x_t}$) as input. Then, it outputs a binary number, which indicates if the local output belongs to the source or target domain. In other words, local discriminator aligns the local outputs across source and target domain. Moreover, for the source domain, local outputs corresponding to the background category can be discarded by leveraging the segmentation ground truth map. So, we do not account adversarial loss for $\boldsymbol{p}_{x_s}^{i,j}$ corresponding to source category outputs belonging to background pixels.

Mixture Module. In the early stage of training, the rep-

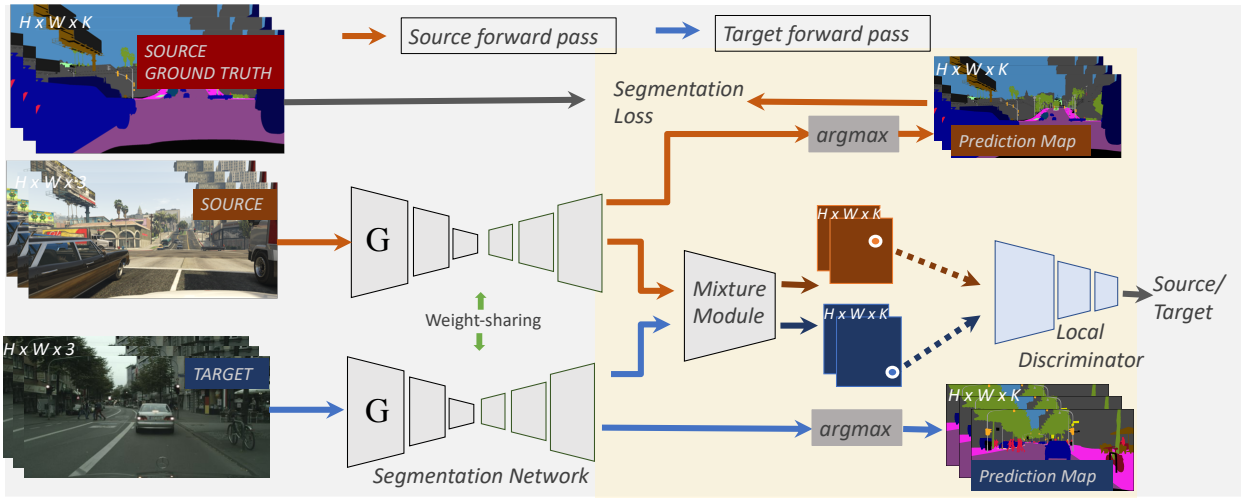

Fig. 2. **Overview of the local adversarial alignment.** It consists of a Segmentation Model $\boldsymbol{G}$ and a local discriminator $\boldsymbol{D}$ along with a mixture module. For an input source image, the output map is used to calculate adversarial loss as well as the segmentation loss. For an input target image, the output map is used to calculate the adversarial loss. Note that the local discriminator performs alignment by discriminating a local output (shown by circles with white border) between source and target domain. The proposed mixture module along with local discriminator represents our contribution of this work.

resentation gap between the output distributions of source and target domain would be large due to domain gap. Methods like [19, 20] have shown that generating intermediate representations in the image space help with the down-stream vision tasks. Inspired by them, in order to fill the representation gap between the two domains, we generate a continuous sequence of intermediate domain local outputs using a mixture module. Mixture Module receives a source output $\boldsymbol{p}_{x_s}$ and target output $\boldsymbol{p}_{x_t}$ as input and produces $\{\boldsymbol{m}_1, \boldsymbol{m}_2\}$, where $\{\boldsymbol{m}_1, \boldsymbol{m}_2\} = f(\boldsymbol{p}_{x_s}, \boldsymbol{p}_{x_t})$, where f is a mapping function. The local outputs at a spatial location from $\boldsymbol{m}_1$, $\boldsymbol{m}_2$ are passed as inputs to the local discriminator. Mixture module is defined as follows:

$$\boldsymbol{m}_1^{(i,j)} = \alpha \boldsymbol{p}_{x_s}^{(i,j)} + (1-\alpha)\boldsymbol{p}_{x_t}^{(i,j)}, \quad l^{\boldsymbol{m}_1^{(i,j)}} = \alpha l^s + (1-\alpha) l^t \tag{2}$$

$$\boldsymbol{m}_2^{(i,j)} = (1-\alpha)\boldsymbol{p}_{x_s}^{(i,j)} + (\alpha)\boldsymbol{p}_{x_t}^{(i,j)}, \quad l^{\boldsymbol{m}_2^{(i,j)}} = (1-\alpha) l^s + (\alpha) l^t \tag{2}$$

Here $\theta < \alpha \leq 1$ and α . l^s is source domain label and l^t is target domain label defined for adversarial training. we use soft label $l^{\boldsymbol{m}_1^{(i,j)}}$ and $l^{\boldsymbol{m}_2^{(i,j)}}$ as ground truth corresponding to the mixed output while training the discriminator. This modifies the adversarial learning objective as follows:

$$\begin{aligned}\mathcal{L}_{local} = \mathbb{E}_{\{\boldsymbol{x}_s \in \boldsymbol{X}_S, \boldsymbol{x}_t \in \boldsymbol{X}_T\}} \sum_{i,j} l^{\boldsymbol{m}_1^{(i,j)}} \log(\boldsymbol{D}(\boldsymbol{m}_1^{(i,j)})) + \\ \sum_{i',j'} l^{\boldsymbol{m}_2^{(i',j')}} \log(1 - \boldsymbol{D}(\boldsymbol{m}_2^{(i',j')}))\end{aligned} \tag{3}$$

3.1. Training Procedure

We follow a two-step training procedure. First, we train our model using global alignment and optimize the target function. $\underset{\boldsymbol{G}}{\text{argmin}} \max_{\boldsymbol{D}} \quad \mathcal{L}_{step1} = \mathcal{L}_{seg}^{src} + \lambda_{adv}(\mathcal{L}_{global})$, where λ_{adv} are the weight parameter for the adversarial losses. Second, after obtaining the model trained to conduct global alignment in the first step, we train it further to optimize the loss function: $\underset{\boldsymbol{G}}{\text{argmin}} \max_{\boldsymbol{D}} \quad \mathcal{L}_{step2} = \mathcal{L}_{seg}^{src} + \lambda_{adv}(\mathcal{L}_{local})$, where λ_{adv} are the weight parameter for the adversarial losses.

3.2. Implementation Details

For a fair comparison, we follow baselines [21, 11] and adopt their segmentation architecture as our segmentation network $\boldsymbol{G}$. Our local discriminator is a stack of 1x1 kernel, Leaky ReLU blocks. Leaky ReLU parameterized by 0.2. We implement the proposed framework using PyTorch toolbox on NVIDIA Tesla V100 GPU with 16 GB memory. In all our experiments, we initialise our backbone network with a model already trained for global alignment. To train the discriminator, we use the Adam Optimizer [22] with learning rate of 10^{-4} and momentum parameters set as 0.9 and 0.99. For training the segmentation network $\boldsymbol{G}$, we use the Stochastic Gradient Decent (SGD) with 2.5×10^{-4} learning rate and momentum as 0.9. We select λ_{adv} = 0.001. We use θ as 0.6, 0.8 for GTA5 to Cityscapes and SYNTHIA to Cityscapes task and train the network with mixture module for initial 30,000 iterations. Later on, we continue to train with θ as 1.0 for another 50,000 iterations. We use $l^s = 0$ and $l^t = 1$.

4. EXPERIMENTAL RESULTS

Datasets. We evaluate our method on two domain adaptation tasks: (1) GTA5 [6] as source to, Cityscapes [23] as the target (**GTA2City**) and (2) SYNTHIA [7] as source to, Cityscapes as the target (**SYN2City**). Cityscapes contains 2,993 images for training and 500 images for validation. GTA5, SYNTHIA contains 24,999 and 9,400 images, respectively. We provide

Method	Publica.	Arch.	GTA2City mIoU(%)	SYN2City mIoU(%)
AdapSegNet [3]	CVPR'18	V	35.0	37.6
Our AdapSegNet	-	V	**36.8** $\pm$ **0.2** **(+1.8)**	**38.9** $\pm$ **0.3** **(+1.3)**
BDL [21]	CVPR'19	V	41.3	46.1
Our BDL	-	V	**43.5** $\pm$ **0.2** **(+2.2)**	**47.7** $\pm$ **0.2** **(+1.6)**
Source only	-	-	36.6	38.6
AdapSegNet [3]	CVPR'18	R	41.4	45.9
Our AdapSegNet	-	R	**45.2** $\pm$ **0.1** **(+3.8)**	**46.9** $\pm$ **0.3** **(+1.0)**
ADVENT [11]	ICCV'19	R	43.8	47.6
Our ADVENT	-	R	**46.4** $\pm$ **0.3** **(+2.6)**	**48.3** $\pm$ **0.4** **(+0.7)**
BDL [21] $M_2^{(2)}(F^{(2)})$	CVPR'19	R	48.5	51.4
Our BDL	-	R	**49.4** $\pm$ **0.2** **(+0.9)**	**52.5** $\pm$ **0.3** **(+1.1)**

Table 1. **Semantic segmentation performance in mIoU(%)** on GTA5 to Cityscapes (GTA2City) and Synthia to Cityscapes (SYN2City) Adaptation task. 'R' means the ResNet-101 and 'V' means the VGG-16 backbone. Our global-local alignment approach shows consistent improvement over baselines.

results over 10 training episodes of the model. We report the results with mean IoU for 19 categories for GTA2City task and 13 categories for SYN2City task.

Integrating with AdapSegNet [3]. We conduct local alignment with globally aligned AdapSegNet [3] as our baseline. As shown in Table 1, our proposed approach improves adaptation of 11/19, 14/19 categories and improves by 3.8% and 1.8% in mean IoU for GTA2City adaptation task on the two architectures, respectively. In Table 1, we present results for SYN2City adaptation task and observe similar improvements. Consistent improvements on majority of the categories validates our intuition that local alignment after global alignment can result in much better alignment.

Integrating with BDL [21]. We utilize BDL [21] that along with global alignment does image-image translation and self-training as our baseline. We use [21] generated GTA5 to Cityscapes and SYNTHIA to Cityscapes image translation results[1] and their trained model to generate the pseudo labels. We conduct local alignment after conducting global alignment. Our method achieves 49.4%, 52.5% in mean IoU and improves adaptation of 13/19, and 9/13 categories (see Table 1) over the baseline with ResNet-101. As shown in Table 1, conducting local alignment on top of global alignment significantly increases the accuracy of tail categories *e.g.,* light (5.5%), sign (4.7%), rider (4.4%), motorbike (4.2%), bicycle (1.4%) which shows the robustness of our approach, and its effectiveness to achieve much better adaptation over a stronger baseline.

4.1. Ablation Studies

In this section, we conduct detailed ablation on how much each component in our approach contributes to the overall accuracy. Table 3 gives the details with 4 training setups S1 to S4: S1 represents use of AdapSegNet [3] which conducts global alignment, S2 amounts to using the local discriminator without mixture module along with S1. S3 amounts to removing adaptation of background source pixels from S2. S4 accounts to using mixture module with S3. Improvement of 1.0% mIoU from S2 to S3 justifies our claim that outputs corresponding to source background pixels hinder adaptation. Improvement of 1.3% mIoU from S3 to S4 indicate the effectiveness of output mixture module.

[1] https://github.com/liyunsheng13/BDL

Method	Arch.	mIoU(%)*	Gain
AdapSegNet [3]	R	31.2	-
Our AdapSegNet	R	**33.5**	**+2.3**

Table 2. **Ablation Study on the importance of local alignment without source outputs corresponding to the background pixels.** Our method improve by 2.3% over AdapSegNet baseline for a total of 17 categories.

Setup	AdapSegNet	Local disc.	No BG adap.	Mixture Mod.	GTA2City mIoU(%)
S1 (Global Alignment)	✓				41.4
S2	✓	✓			42.9
S3	✓	✓	✓		43.9
S4 (Global-Local Alignment)	✓	✓	✓	✓	**45.2**

Table 3. **Ablation studies on contribution of each component in our approach.** Here 'No BG adap.' means excluding source background pixels from adaptation. Results of GTA2City in % mIoU for 19 categories. We notice consistent improvements from various modules used in our approach.

Excluding background outputs from adaption. We design an experiment to study the impact of adapting source local outputs corresponding to the background pixels and demonstrate how our model can alleviate it. We take GTA5 as source domain which contains 13.6%, 16.7% of total pixels belonging to 'sky' and 'building' categories, and reassign them as background pixels, which makes a total of 41.1% (including original 11%) background pixels. We conduct GTA5 to Cityscapes domain adaptation with AdapSegNet [3] and conduct local alignment on top of AdapSegNet that already does global alignment. As shown in Table 2, conducting local alignment is able to better alleviate the effect of background pixels on adaptation and improves mIoU over AdapSegNet by 2.3% for a total of 17 categories.

5. CONCLUSION

We propose a global-local adversarial learning framework for semantic segmentation. Instead of only conducting global alignment, we show that conducting local alignment following a global alignment better adapts local outputs for unsupervised domain adaptation. Our global-local alignment results on benchmark semantic segmentation datasets demonstrate the effectiveness of our approach.

Acknowledgement. This work is supported in part by a gift grant from Horizon Robotics and National Science Foundation Grant CNS1951952.

6. REFERENCES

[1] Yaroslav Ganin and Victor Lempitsky, "Unsupervised domain adaptation by backpropagation," *arXiv preprint arXiv:1409.7495*, 2014.

[2] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell, "Adversarial discriminative domain adaptation," in *CVPR*, 2017, pp. 7167–7176.

[3] Yi-Hsuan Tsai, Wei-Chih Hung, Samuel Schulter, Kihyuk Sohn, Ming-Hsuan Yang, and Manmohan Chandraker, "Learning to adapt structured output space for semantic segmentation," in *CVPR*, 2018, pp. 7472–7481.

[4] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[5] Shuang Li, Chi Harold Liu, Binhui Xie, Limin Su, Zhengming Ding, and Gao Huang, "Joint adversarial domain adaptation," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 729–737.

[6] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun, "Playing for data: Ground truth from computer games," in *European conference on computer vision*. Springer, 2016, pp. 102–118.

[7] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *CVPR*, 2016, pp. 3234–3243.

[8] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros, "Image-to-image translation with conditional adversarial networks," in *CVPR*, 2017, pp. 1125–1134.

[9] Swami Sankaranarayanan, Yogesh Balaji, Arpit Jain, Ser Nam Lim, and Rama Chellappa, "Learning from synthetic data: Addressing domain shift for semantic segmentation," in *CVPR*, 2018, pp. 3752–3761.

[10] Zak Murez, Soheil Kolouri, David Kriegman, Ravi Ramamoorthi, and Kyungnam Kim, "Image to image translation for domain adaptation," in *CVPR*, 2018, pp. 4500–4509.

[11] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez, "Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation," in *CVPR*, 2019, pp. 2517–2526.

[12] Weixiang Hong, Zhenzhen Wang, Ming Yang, and Junsong Yuan, "Conditional generative adversarial network for structured domain adaptation," in *CVPR*, 2018, pp. 1335–1344.

[13] Yiheng Zhang, Zhaofan Qiu, Ting Yao, Dong Liu, and Tao Mei, "Fully convolutional adaptation networks for semantic segmentation," in *CVPR*, 2018, pp. 6810–6818.

[14] Yun-Chun Chen, Yen-Yu Lin, Ming-Hsuan Yang, and Jia-Bin Huang, "Crdoco: Pixel-level domain transfer with cross-domain consistency," in *CVPR*, 2019, pp. 1791–1800.

[15] Junbao Zhuo, Shuhui Wang, Weigang Zhang, and Qingming Huang, "Deep unsupervised convolutional domain adaptation," in *Proceedings of the 25th ACM international conference on Multimedia*, 2017, pp. 261–269.

[16] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell, "Cycada: Cycle-consistent adversarial domain adaptation," in *International conference on machine learning*. PMLR, 2018, pp. 1989–1998.

[17] Yawei Luo, Liang Zheng, Tao Guan, Junqing Yu, and Yi Yang, "Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation," in *CVPR*, 2019, pp. 2507–2516.

[18] Haoran Wang, Tong Shen, Wei Zhang, Lingyu Duan, and Tao Mei, "Classes matter: A fine-grained adversarial approach to cross-domain semantic segmentation," *arXiv preprint arXiv:2007.09222*, 2020.

[19] Rui Gong, Wen Li, Yuhua Chen, and Luc Van Gool, "Dlow: Domain flow for adaptation and generalization," in *CVPR*, 2019, pp. 2477–2486.

[20] Minghao Xu, Jian Zhang, Bingbing Ni, Teng Li, Chengjie Wang, Qi Tian, and Wenjun Zhang, "Adversarial domain adaptation with domain mixup," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, vol. 34, pp. 6502–6509.

[21] Yunsheng Li, Lu Yuan, and Nuno Vasconcelos, "Bidirectional learning for domain adaptation of semantic segmentation," in *CVPR*, 2019, pp. 6936–6945.

[22] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[23] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele, "The cityscapes dataset for semantic urban scene understanding," in *CVPR*, 2016, pp. 3213–3223.