Unsupervised Domain Adaptation for Object Detection via Cross-Domain Semi-Supervised Learning

Fuxun Yu[†], Di Wang[‡], Yinpeng Chen[‡], Nikolaos Karianakis[‡], Pei Yu[‡], Dimitrios Lymberopoulos[‡], Xiang Chen[†]
[†] George Mason University, [‡] Microsoft
[†]{fyu2, xchen26}@gmu.edu, [‡]{wangdi, yiche, nikolaos.karianakis, pei.yu, dlymper}@microsoft.com

Abstract

Unsupervised Domain Adaptation (UDA) is a promising approach to adapt models for new domains/environments. Previously, many adversarial methods are proposed to conduct feature alignment for adaptation. However, such adversarial-based methods can only reduce domain style gap, but cannot address the domain content distribution gap that is also important for object detectors. To overcome this limitation, we propose the Cross-Domain Semi-Supervised Learning (CDSSL) framework by leveraging high quality pseudo labels to learn from target domain directly. Meanwhile, we conduct fine-grained domain transfer to reduce the style gap. Experiments show our approach achieves new state-of-the-art performance (2.2% - 9.5% better than the best prior work on mAP). The full paper could be found at https://arxiv.org/abs/1911.07158. Code will be available at https://github.com/Mrxiaoyuer/CDSSL.

1. Introduction

Recently, Unsupervised Domain Adaptation (UDA) has become a promising approach to adapt model into dynamic real-world scenarios [2, 5, 9, 10]. Previously, most UDA works for object detectors [1, 13, 7, 11] use adversarial feature alignment methods to learn invariant features from source/target domains. However, due to the lack of target-domain labels, adversarial methods can only perform coarse-grained feature alignment and are prone to misalignment, e.g., misaligning features of fore-/backgrounds, different classes, etc. Meanwhile, they may not be able to align content distribution shift between two domains [12].

Targeting at these issues, we take a different adaptation approach by a semi-supervised learning (SSL) method: *Self-Training* [6]. Self-Training utilizes labeled data to train annotators and generates *pseudo labels* on unlabeled data. Both parts of data are then combined for further training [4]. Although such *pseudo labels* are not as accurate as ground-truth (GT) labels, they bring several benefits compared to adversarial feature alignment method: (i) they enabled detectors to learn detection loss from target-domain images directly, instead of in-directly being aligned by discriminator in feature level; (ii) the overall pseudo labels on the target domain approximates the real data distribution, reducing the potential content distribution gaps.

We evaluate our CDSSL framework on several detection adaptation benchmarks, including *synthetic-to-real*, *cross-camera*, and *normal-to-foggy*. Our approach performs consistently better than prior best work by 2.2% - 9.5% mAP, achieving the new SOTA detection adaptation performance.

2. Methodology

CDSSL Framework Overview. The CDSSL framework consists of two major steps: **(a)** To reduce the domain style gaps, we first conduct *fine-grained domain style transfer*. Naive CycleGAN tends to modify objects during translation due to large receptive field. Therefore, we restrict its receptive filed so that it only translates styles and better reserves small objects. Then the initial pseudo label annotator will be trained on the style-translated domain to generate initial pseudo labels on target domain. **(b)** We then run *iterative self-training* by combing the source-domain data (with ground-truth (GT) labels) and the target-domain data (with pseudo labels) together. Inevitably, the pseudo labels contain some annotation errors. Therefore, we conduct *imbalanced mini-batch sampling* per training iteration, to under-sample the pseudo labels and over-sample the GT labels. After each self-training round, we use the better-trained models to conduct *confidence-based hard labeling* for better pseudo labels. The self-training could iterate multiple times till the model performance stops improving.

Fine-grained Domain Style Transfer Original cyclegan design does not target at any specific end tasks like object detection. As a result, vanilla cyclegan often not only translates the styles, but also translates major objects or backgrounds (Fig. 1 (a)), which is detrimental for detection tasks. We find this is due to the large receptive field of cyclegan, which enables it to learn translating entire objects.

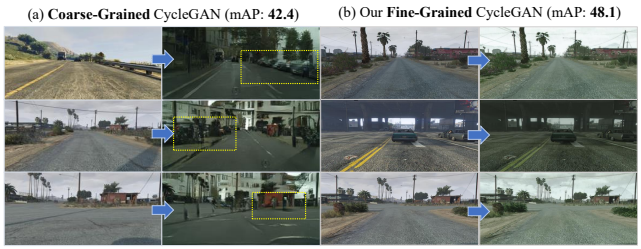

Figure 1: Coarse-grained v.s. our fine-grained CycleGAN.

To solve this problem, we propose a simple yet effective method: *restricting the receptive field of cyclegan.* Specifically, in training process, we restrict the translating patch size of the generator and discriminator so that the models can only learn translating the local textures without "seeing" any major objects or contexts. During testing, by using the fully-convolutional model structure, the generator can be applied on the full-size image for fine-grained style translation. Fig. 1 (a) and (b) compare the results between coarse-/fine-grained translation. Clearly, the cyclegan with large receptive field are not suitable for detection adaptation with even missing objects during translation. By contrast, our fine-grained one translates styles and preserves objects both well. On the *syn-to-real* benchmark, such fine-grained design alone leads to +5.7% mAP for the following detection tasks than naive cyclegan translation.

Iterative Self-Training After domain transfer, we conduct initial annotator training on the style-translated source domain. Then the target domain is annotated with pseudo labels by the annotator. Such pseudo labels on target domain can contain some errors. Therefore, we optimize the self-training with imbalanced sampling per iteration and confidence-based labeling per self-training round.

Imbalanced Sampling: Since the source-domain samples have ground-truth labels, the loss from source labels can be more accurate, especially in the localization head. Therefore, we over-sample the source domain samples but under-sample the target domain ones during training, which provides training stabilization and error-correction effects.

Confidence-based Labeling: At the end of each self-training round, we update the pseudo labels by applying the better-trained detector on the target domain. But to control the label quality, we use confidence-thresholding to choose most confident predictions and sharpen the soft probability into hard labels to learn more confident representations: During iterative self-training rounds, some error boxes' confidence will be reinforced and similar wrong predictions can appear later with higher confidence. Therefore, we also progressively increase the confidence threshold in later rounds.

3. Experimental Evaluation

Experiments Setup. For experiments setup, we follow the same settings as in [1, 13, 7]. We use Faster-RCNN with VGG16 backbone. Three adaptation scenarios are evaluated, namely Synthetic to Real (*Sim2City*), Cross Camera (*Kitti2City*) and Normal to Foggy (*City2Foggy*). For fair comparison with prior works, we evaluate our framework under two resolution settings: 512 pixels or 600 pixels as image's shorter side. We report the mAP at IoU=0.5.

Table 1: Sim10k to Cityscapes (Resolution: 512 & 600).

Methods	*Car AP*
^{512}Baseline	33.0
^{512}CVPR'18 [1]	39.0
^{512}CVPR'19 [13]	43.0
^{512}**Ours**	**49.0**
^{600}Baseline	34.6
^{600}CVPR'19 [7]	42.3
^{600}ICCV'19 [11]	42.8
^{600}**Ours**	**52.3**

Table 2: KITTI to Cityscapes (Resolution: 512 & 600).

Methods	*Car AP*
^{512}Baseline	36.4
^{512}CVPR'18 [1]	38.5
^{512}CVPR'19 [13]	42.5
^{512}**Ours**	**45.2**
^{600}Baseline	37.5
^{600}CVPR'19 [7]	-
^{600}ICCV'19 [11]	-
^{600}**Ours**	**46.4**

Table 3: Cityscapes to Foggy-Cityscapes Performance.

Methods/Class	1	2	3	4	5	6	7	8	mAP
^{512}Baseline	29.7	32.2	44.6	16.2	27.0	9.1	20.7	29.7	26.2
^{512}CVPR'18 [1]	25.0	31.0	40.5	22.1	35.3	20.2	20.0	27.1	27.6
^{512}CVPR'19 [13]	33.5	38.0	48.5	26.5	39.0	23.3	28.0	33.6	33.8
^{512}**Ours**	33.9	38.7	52.1	26.3	43.4	**32.9**	27.5	35.5	**36.3**
^{600}CVPR'19 [7]	29.9	42.3	43.5	24.5	36.2	32.6	30.0	35.3	34.3
^{600}ICCV'19 [11]	33.2	**44.2**	44.8	**28.2**	41.8	28.7	30.5	36.5	36.0
^{600}**Ours**	**38.2**	42.1	**55.6**	25.9	**43.5**	27.6	**33.5**	**39.2**	**38.2**

Synthetic to Real Adaptation. Here we use SIM10K → Cityscapes. The results are shown in Table 1 with baseline and other methods [1, 13, 7, 11]. Compared to prior works, our approach achieves the new SOTA performance, **+6.0%** and **+9.5%** than prior work in both 512 and 600 resolutions.

Cross Camera Adaptation. Here we use KITTI → Cityscapes [3]. The results are shown in Table 2. Compared to baseline, our method brings **+8.8%** improvement, and outperforms the previous best result [13] by **+2.7%**.

Multi-Class Normal to Foggy Adaptation. In this part, we evaluate our framework on Cityscapes → Foggy-Cityscapes [8]. As shown in Table 3, our approach achieves the best performance, achieving **+2.5%** and **+2.2%** mAP gain compared to prior SOTA performance [13].

4. Conclusion

In this work, we propose CDSSL: a cross-domain semi-supervised learning framework to address the UDA problem for object detection. Specifically, we conduct domain transfer and then launch the iterative self-training. Imbalanced sampling and confidence-based label sharpening are also proposed to mitigate the label errors. Experiments show that our work consistently outperforms previous SOTA by 2.2% - 9.5% in various adaptation scenarios.

References

[1] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3339–3348, 2018. 1, 2

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

[3] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 2

[4] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, ICML*, volume 3, page 2, 2013. 1

[5] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Unsupervised domain adaptation with residual transfer networks. In *Advances in Neural Information Processing Systems*, pages 136–144, 2016. 1

[6] David McClosky, Eugene Charniak, and Mark Johnson. Effective self-training for parsing. In *Proceedings of the main conference on human language technology conference of the North American Chapter of the Association of Computational Linguistics*, pages 152–159. Association for Computational Linguistics, 2006. 1

[7] Kuniaki Saito, Yoshitaka Ushiku, Tatsuya Harada, and Kate Saenko. Strong-weak distribution alignment for adaptive object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6956–6965, 2019. 1, 2

[8] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Semantic foggy scene understanding with synthetic data. *International Journal of Computer Vision*, 126(9):973–992, 2018. 2

[9] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7167–7176, 2017. 1

[10] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*, 2014. 1

[11] Rongchang Xie, Fei Yu, Jiachao Wang, Yizhou Wang, and Li Zhang. Multi-level domain adaptive learning for cross-domain detection. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 1, 2

[12] Han Zhao, Remi Tachet des Combes, Kun Zhang, and Geoffrey J Gordon. On learning invariant representation for domain adaptation. *arXiv preprint arXiv:1901.09453*, 2019. 1

[13] Xinge Zhu, Jiangmiao Pang, Ceyuan Yang, Jianping Shi, and Dahua Lin. Adapting object detectors via selective cross-domain alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 687–696, 2019. 1, 2