Active Domain Adaptation with False Negative Prediction for Object Detection

Supplementary Material

S1. Details of Formulation

We detail the formulation described in Section 3.

Notation. We have a set of labeled source data $\mathcal{D}_S = \{(\boldsymbol{x}_i^S, \boldsymbol{y}_i^S)\}_{i=1}^{N_S}$ and unlabeled target data $\mathcal{D}_T = \{\boldsymbol{x}_i^T\}_{i=1}^{N_T}$, where $\boldsymbol{x}_i \in \mathbb{R}^{W\times H\times 3}$ is the i-th image of width W and height H in a dataset, and $\boldsymbol{y}_i = \{\boldsymbol{b}_{i,j}, c_{i,j}\}_{j=1}^{N_{bbox}^i}$ consists of the j-th bounding box coordinates $\boldsymbol{b}_{i,j} \in \{x, y, w, h\}$ and category index $c_{i,j} \in \{1, ..., N_c\}$. $\mathcal{D}_T$ consists of a set of labeled target data $\mathcal{D}_{LT} = \{(\boldsymbol{x}_i^{LT}, \boldsymbol{y}_i^{LT})\}_{i=1}^{N_{LT}}$ and unlabeled target data $\mathcal{D}_{UT} = \{\boldsymbol{x}_i^{UT}\}_{i=1}^{N_{UT}}$. Given $\mathcal{D}_{LT} = \emptyset$ at the beginning of training, we sample images from $\mathcal{D}_{UT}$ that maximize performance through an acquisition function within a labeling budget. Then, we annotate these images and incorporate them into $\mathcal{D}_{LT}$.

S1.1. Model Initialization

Given the detection model and domain discriminator parameters θ_s and ϕ_s, respectively, of the student model, the objective loss function is defined as follows:

$$\min_{\theta_s}\max_{\phi_s} \mathcal{L}_{init} = \mathcal{L}_{sup}^S + \lambda\mathcal{L}_{adv}, \tag{S1}$$

where λ is the weight of $\mathcal{L}_{adv}$. $\mathcal{L}_{sup}^S$ is the supervised loss in the source domain defined as follows:

$$\mathcal{L}_{sup}^S = \frac{1}{N_S}\sum_{i=1}^{N_S}\mathcal{L}_{det}(\boldsymbol{x}_i'^S, \boldsymbol{y}_i'^S), \tag{S2}$$

where $\{\boldsymbol{x}', \boldsymbol{y}'\}$ is the augmented sample by strong augmentation proposed in [2]. $\mathcal{L}_{det}$ is the detection loss defined as follows:

$$\mathcal{L}_{det}(\boldsymbol{x}_i, \boldsymbol{y}_i) = \frac{1}{N_{bbox}^i}\sum_{j=1}^{N_{bbox}^i}(\mathcal{L}_{cls}^{rpn}(\boldsymbol{x}_i, c_{i,j}) + \mathcal{L}_{reg}^{rpn}(\boldsymbol{x}_i, \boldsymbol{b}_{i,j}) + \mathcal{L}_{cls}^{roi}(\boldsymbol{x}_i, c_{i,j}) + \mathcal{L}_{reg}^{roi}(\boldsymbol{x}_i, \boldsymbol{b}_{i,j})), \tag{S3}$$

where N_{bbox}^i is the number of bounding boxes in $\boldsymbol{y}_i$. $\mathcal{L}_{cls}^{rpn}$ and $\mathcal{L}_{reg}^{rpn}$ are the classification loss and regression loss, respectively, in the region proposal network (RPN). $\mathcal{L}_{cls}^{roi}$ and $\mathcal{L}_{reg}^{roi}$ are the classification loss and regression loss, respectively, in the region of interest (RoI) head.

The adversarial loss $\mathcal{L}_{adv}$ is defined as follows:

$$\mathcal{L}_{adv} = -\frac{1}{N_S}\sum_{i=1}^{N_S}\log(1 - D(F_{enc}(\boldsymbol{x}_i^S;\theta_s);\phi_s)) - \frac{1}{N_T}\sum_{i=1}^{N_T}\log D(F_{enc}(\boldsymbol{x}_i^T;\theta_s);\phi_s), \tag{S4}$$

where F_{enc} and D are the backbone of the detection model and the domain discriminator, respectively.

S1.2. Active Learning based on False Negative

S1.2.1 False Negative Prediction Module

The FNPM is trained to minimize errors using the loss function, which is defined as follows:

$$\mathcal{L}_{fn} = \frac{1}{N_S}\sum_{i=1}^{N_S}(G(F_{enc}(\boldsymbol{x}_i^S;\theta_t);\psi) - \mathcal{FN}(F_{head}(\boldsymbol{x}_i^S;\theta_t), \boldsymbol{y}_i^S))^2 + \frac{1}{N_{LT}}\sum_{i=1}^{N_{LT}}(G(F_{enc}(\boldsymbol{x}_i^{LT};\theta_t);\psi) - \mathcal{FN}(F_{head}(\boldsymbol{x}_i^{LT};\theta_t), \boldsymbol{y}_i^{LT}))^2, \tag{S5}$$

where G, ψ, and F_{head} are the FNPM, its parameters, and the head of the detection model, respectively.

S1.2.2 Uncertainty Estimation with MCDropout

We incorporate the MCDropout layer into the detection head and reformulate the predicted bounding box coordinates $\hat{\boldsymbol{b}}_i$ and class probabilities $\hat{\boldsymbol{p}}_i$ as follows:

$$\{\hat{\boldsymbol{b}}_i(\xi), \hat{\boldsymbol{p}}_i(\xi)\} = F_{head}(\boldsymbol{x}_i;\theta_t,\xi), \text{ where } \xi \sim Ber(\eta), \tag{S6}$$

where $Ber(\eta)$ is Bernoulli distribution of dropout rate η.

We use these means $(\hat{\boldsymbol{b}}_i^{mean}, \hat{\boldsymbol{p}}_i^{mean})$ as the prediction results considering model perturbations and the variance of the predicted coordinates $\hat{\boldsymbol{b}}_i^{var}$ as the localization uncertainty defined as follows:

$$\hat{\boldsymbol{b}}_i^{mean} = \frac{1}{M}\sum_{m=1}^{M}\hat{\boldsymbol{b}}_{i,m}, \quad \hat{\boldsymbol{p}}_i^{mean} = \frac{1}{M}\sum_{m=1}^{M}\hat{\boldsymbol{p}}_{i,m}, \tag{S7}$$

$$\hat{\boldsymbol{b}}_i^{var} = \frac{1}{M-1}\sum_{m=1}^{M}(\hat{\boldsymbol{b}}_{i,m} - \hat{\boldsymbol{b}}_i^{mean})^2, \tag{S8}$$

where M is the number of inferences and $\hat{\boldsymbol{b}}_{i,m} \sim \hat{\boldsymbol{b}}_i(\xi)$, $\hat{\boldsymbol{p}}_{i,m} \sim \hat{\boldsymbol{p}}_i(\xi)$.

S1.3. Semi-Supervised Domain Adaptation

The objective loss function is defined as follows:

$$\min_{\theta_s}\max_{\phi_s} \mathcal{L}_{total} = \mathcal{L}_{sup}^S + \mathcal{L}_{sup}^{LT} + \mathcal{L}_{unsup} + \lambda\mathcal{L}_{adv}, \tag{S9}$$

where $\mathcal{L}_{sup}^{LT}$ is the supervised loss in the labeled target domain, similar to Eq. (S2). $\mathcal{L}_{unsup}$ is the unsupervised loss in the unlabeled target domain defined as follows:

$$\mathcal{L}_{unsup} = \frac{1}{N_{UT}} \sum_{i=1}^{N_{UT}} \frac{1}{N_{bbox}^{i}} \sum_{j=1}^{N_{bbox}^{i}} \mathbb{I}_{bbox}(\hat{\boldsymbol{b}}_{i,j}^{var}) \mathbb{I}_{conf}(\hat{\boldsymbol{p}}_{i,j}^{mean}) [\mathcal{L}_{cls}^{rpn}(\boldsymbol{x}_i'^{UT}, c_{i,j}^{PL}) + \mathcal{L}_{cls}^{roi}(\boldsymbol{x}_i'^{UT}, c_{i,j}^{PL})], \tag{S10}$$

where $\mathbb{I}_{bbox}(\cdot)$ is the indicator function defined as follows:

$$\mathbb{I}_{bbox}(\hat{\boldsymbol{b}}_{i,j}^{var}) = \begin{cases} 1, & \text{if } \frac{1}{4} \sum_{k \in \{x,y,w,h\}} \hat{b}_{i,j,k}^{var} \leq \gamma \\ 0, & \text{otherwise,} \end{cases} \tag{S11}$$

where γ is the threshold for using pseudo-labels with variance less than a certain value. $\mathbb{I}_{conf}(\cdot)$ is the indicator function defined as follows:

$$\mathbb{I}_{conf}(\hat{\boldsymbol{p}}_{i,j}^{mean}) = \begin{cases} 1, & \text{if } max(\hat{\boldsymbol{p}}_{i,j}^{mean}) \geq \tau \\ 0, & \text{otherwise,} \end{cases} \tag{S12}$$

where τ is the threshold for using pseudo-labels with a maximum class probability above a specific value. Pseudo-labels are defined as follows:

$$c_{i,j}^{PL} = argmax(\hat{\boldsymbol{p}}_{i,j}^{mean}). \tag{S13}$$

Finally, after the student model is updated once by Eq. (S9), the teacher model is updated by using the exponential moving average (EMA) [12]:

$$\theta_t \leftarrow \alpha\theta_t + (1-\alpha)\theta_s, \quad \phi_t \leftarrow \alpha\phi_t + (1-\alpha)\phi_t, \tag{S14}$$

where α is the update ratio.

S2. Additional Results

S2.1. Comparison with Domain Adaptation Methods for Each Category

Tables S1 and S2 show the comparison of accuracies for each category for **Cityscapes** $\rightarrow$ **Foggy Cityscapes** and **Cityscapes** $\rightarrow$ **BDD100k**, respectively, in Table 1. The value in the table are the average precision at IoU=0.5.

In particular, in the bus column of Table S2, our proposed method improves the accuracy by 14.3 to 17.8 pt from AADA. This is because the accuracy of a minority of categories is reflected as undetectability, and the active sampling focuses on images that contain objects in those categories. In Table S1, the accuracy of our proposed method is already higher than that of Oracle, and the accuracy of our proposed method is competitive with the previous method.

S2.2. Comparison with Active Sampling Strategies for Each Category

Table S3 shows the comparison of accuracies for each category for **Cityscapes** $\rightarrow$ **BDD100k** in Table 2. The addition of s^{fn} improves accuracy of person and car and samples more informative images due to the effectiveness of undetectability.

S2.2.1 Analysis of Hyper-Parameters

Table S4 shows the comparison of performance at value of dropout rate η and threshold value γ of pseudo-label with variance in the **KITTI** $\rightarrow$ **Cityscapes** scenario. Changing η has little effect on accuracy. In addition, if γ is set too small, training becomes unstable. Therefore, we set η to 0.1 and γ to 0.1.

S2.2.2 Visualization of Active Sampling

Figure S1 shows top 10 images sampled from the target domain at round 1 in the **Cityscapes** $\rightarrow$ **BDD100k** scenario. We compare the images sampled by the AADA method with those sampled by our proposed method. In AADA, images with poor visibility due to rain on the glass in rainy weather are selected. These images are expected to be hard to detect, however this situation rarely occurs in the target domain, and the images are not effective for training. In contrast, our proposed method can sample more important images with high detection difficulty and high frequency of occurrence. In particular, images containing buses with low accuracy have been selected and are effective for training.

Figure S2 shows statistical comparison of sampled objects and attributes in the **Cityscapes** $\rightarrow$ **BDD100k** scenario. As shown in Figure S2a, in terms of the total number of objects in the selected images, our proposed method included a thousand more objects than AADA. Therefore, our proposed method selects more informative images. In Figure S2b, AADA samples the same amount of images in rainy as in clear weather. In contrast, our proposed method samples mainly clear weather, while other weather conditions are sampled evenly.

S2.2.3 Computational Cost

Table S5 shows a comparison of the time required for training the detection model, training the FNPM, and active sampling. The GPU hours are measured on the NVIDIA V100 GPU. Our proposed method requires additional time to train the FNPM before active sampling and additional time to infer the FNPM model during active sampling. The FNPM does not affect the training time of the detection model because it is trained before active sampling. The additional time during the FNPM inference is negligible, and the total additional time is within 10%.

S3. Limitation

Our proposed method was evaluated on the in-vehicle camera dataset, although it has not been tested on other datasets. We will experiment with adaptation scenarios from real images to illustrative images. Although we evaluated our proposed method on a two-stage detector, it can also be applied

Table S1. **Comparison of state-of-the-art UDA and ADA methods for each category in the Cityscapes → Foggy Cityscapes scenario.** The best and the second best accuracies are highlighted in **bold**.

Methods		Reference	Architecture	Backbone	person	rider	car	truck	bus	train	motor	bike	mAP
Source-only		—	Faster-RCNN	VGG16	20.6	23.9	24.6	6.0	13.1	5.0	9.5	21.3	15.5
UDA	DA-Faster [3]	CVPR'18	Faster-RCNN	VGG16	25.0	31.0	40.5	22.1	35.3	20.2	20.0	27.1	27.6
	SWDA [10]	CVPR'19			29.9	42.3	43.5	24.5	36.2	32.6	30.0	35.3	34.3
	PT [2]	ICML'22			40.2	48.8	59.7	30.7	51.8	30.6	35.4	44.5	42.7
	AT [9]	CVPR'22			45.5	55.1	54.3	35.0	56.3	54.3	38.5	51.9	50.9
	MGADA [15]	CVPR'22			43.9	49.6	60.6	29.6	50.7	39.0	38.3	42.8	44.3
	CMT [1]	CVPR'23			45.9	55.7	63.7	39.6	**66.0**	38.8	41.4	51.2	50.3
	CSDA [4]	ICCV'23			43.1	58.4	44.7	**50.0**	33.3	37.3	42.9	50.5	45.0
	NSA-UDA [16]	ICCV'23			50.2	**60.1**	**67.7**	37.4	57.4	46.9	**47.3**	54.3	**52.7**
	Ours (0%)	—			**50.8**	56.5	67.5	28.1	58.1	**55.6**	44.3	**55.8**	52.0
ADA	AADA† [11] (1%)	WACV'20	Faster-RCNN	VGG16	40.0	43.8	56.0	25.7	38.5	30.6	26.3	36.6	37.2
	AADA† (5%)				42.8	47.1	58.6	21.7	44.4	22.6	31.5	39.7	38.5
	Ours (1%)	—			**50.7**	55.5	67.0	**35.2**	57.2	**53.8**	**43.5**	**54.2**	**52.2**
	Ours (5%)				**50.7**	**56.2**	**68.2**	33.7	**61.2**	50.2	40.5	53.5	51.8
Oracle		—	Faster-RCNN	VGG16	46.4	50.7	65.6	34.9	53.6	38.6	35.1	46.6	46.3

Table S2. **Comparison of state-of-the-art UDA and ADA methods for each category in the Cityscapes → BDD100k scenario.** The best and the second best accuracies are highlighted in **bold**.

Methods		Reference	Architecture	Backbone	person	rider	car	truck	bus	motor	bike	mAP
Source-only		—	Faster-RCNN	VGG16	37.9	28.4	56.6	13.4	15.2	17.3	20.6	26.8
UDA	PDA [6]	WACV'20	Faster-RCNN	VGG16	37.6	32.9	51.8	19.3	23.7	16.1	25.3	29.5
	ICR-CCR [14]	CVPR'20			31.4	31.3	46.3	19.5	18.9	17.3	23.8	26.9
	SFOD [8]	AAAI'21			32.4	32.6	50.4	20.6	23.4	18.9	25.0	29.0
	ILLUME [7]	WACV'22			33.2	20.5	47.8	20.8	**33.8**	24.4	26.7	29.6
	TDD [5]	CVPR'22			39.6	**38.9**	53.9	24.1	25.5	24.5	28.8	33.6
	PT [2]	ICML'22			—	—	—	—	—	—	—	34.9
	NSA-UDA [16]	ICCV'23			—	—	—	—	—	—	—	35.5
	Ours (0%)	—			**50.1**	38.3	**67.1**	**28.3**	5.8	**28.3**	**36.3**	**36.3**
ADA	AADA† [11] (1%)	WACV'20	Faster-RCNN	VGG16	45.7	31.7	60.8	19.8	23.6	20.2	30.0	33.1
	AADA† (5%)				49.8	35.1	65.6	35.2	37.2	29.6	34.6	41.0
	Ours (1%)	—			54.7	41.9	70.5	36.3	41.4	30.8	40.2	45.1
	Ours (5%)				**57.1**	**46.6**	**72.2**	**47.6**	**51.5**	**38.0**	**43.2**	**50.9**
Oracle		—	Faster-RCNN	VGG16	54.5	44.9	73.9	57.6	54.4	38.1	43.0	52.3

to an one-stage detector. In the future, we will also evaluate the effectiveness of an one-stage detector.

References

[1] Shengcao Cao, Dhiraj Joshi, Liang-Yan Gui, and Yu-Xiong Wang. Contrastive mean teacher for domain adaptive object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23839–23848, 2023. 3

[2] Meilin Chen, Weijie Chen, Shicai Yang, Jie Song, Xinchao Wang, Lei Zhang, Yunfeng Yan, Donglian Qi, Yueting Zhuang, Di Xie, and Shiliang Pu. Learning domain adaptive object detection with probabilistic teacher. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, pages 3040–3055, 2022. 1, 3

[3] 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 (CVPR)*, pages 3339–3348, 2018. 3

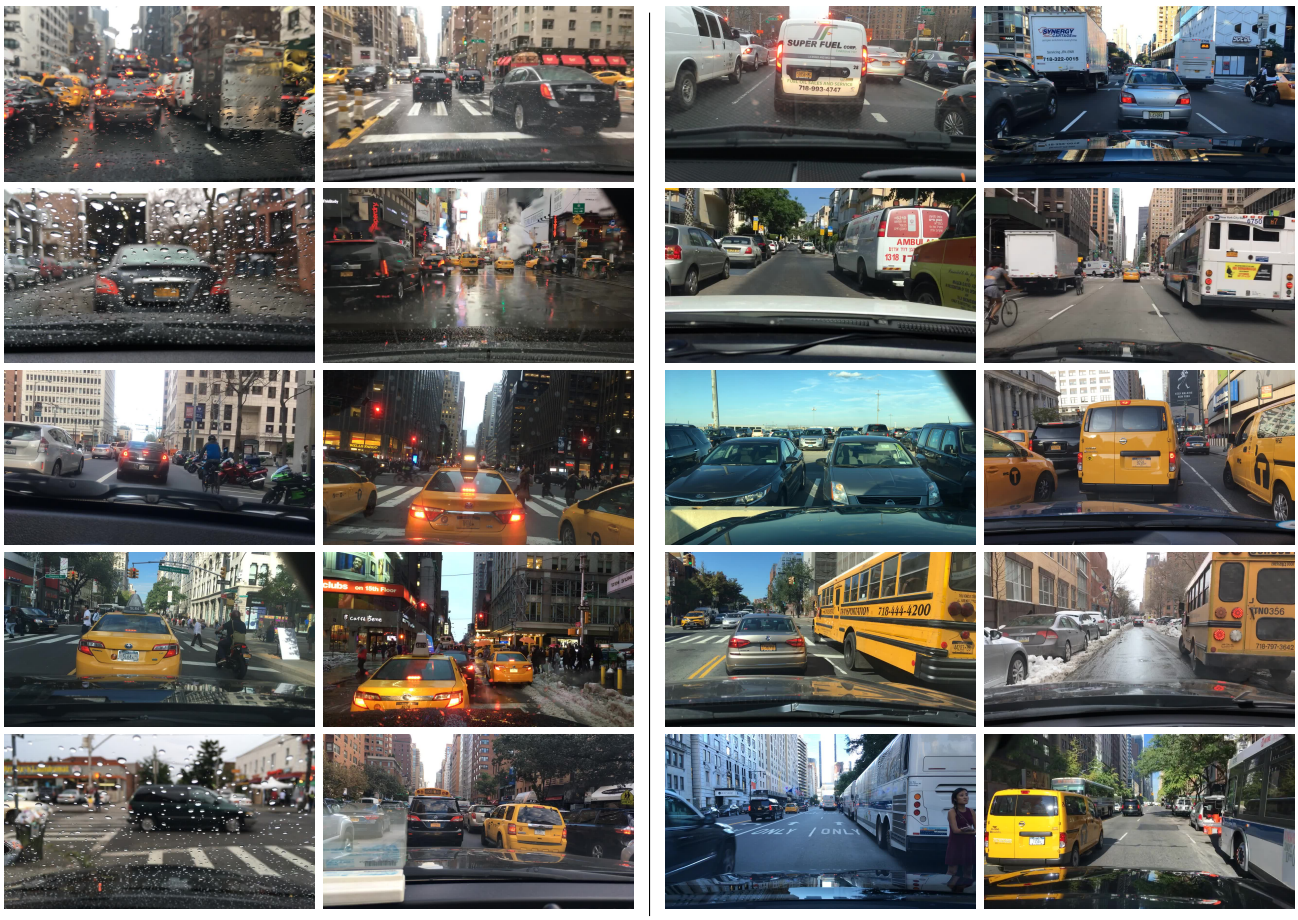
[4] Changlong Gao, Chengxu Liu, Yujie Dun, and Xueming Qian. Csda: Learning category-scale joint feature for domain adaptive object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11421–11430, 2023. 3

[5] Mengzhe He, Yali Wang, Jiaxi Wu, Yiru Wang, Hanqing Li, Bo Li, Weihao Gan, Wei Wu, and Yu Qiao. Cross domain object detection by target-perceived dual branch distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9570–9580, 2022. 3

[6] Han-Kai Hsu, Chun-Han Yao, Yi-Hsuan Tsai, Wei-Chih Hung, Hung-Yu Tseng, Maneesh Singh, and Ming-Hsuan

Table S3. **Comparison of the performances of our proposed method under varying active sampling strategies for each category in the Cityscapes → BDD100k scenario.** The most accurate number is represented in bold. In MeanEntropy and AADA, we underline the results with higher accuracy for the with and without s^{fn}.

Methods	Budget (%)	Architecture	Backbone	person	rider	car	truck	bus	motor	bike	mAP
Random	0.5	Faster-RCNN	VGG16	52.9	39.1	70.0	24.3	12.3	26.0	36.9	37.4
MeanEntropy [13]	0.5	Faster-RCNN	VGG16	53.4	**42.8**	69.4	26.4	20.1	30.3	**40.9**	40.5
MeanEntropy w/ s^{fn}				**54.2**	42.1	**70.1**	25.1	17.3	28.4	40.7	41.7
AADA [11]	0.5	Faster-RCNN	VGG16	52.6	39.5	69.3	29.5	19.6	**30.6**	39.2	40.1
AADA w/ s^{fn}				53.6	40.9	69.8	**34.2**	18.5	29.3	38.5	40.7
Ours	0.5	Faster-RCNN	VGG16	53.7	40.7	69.9	32.2	**35.9**	28.2	37.3	**42.6**



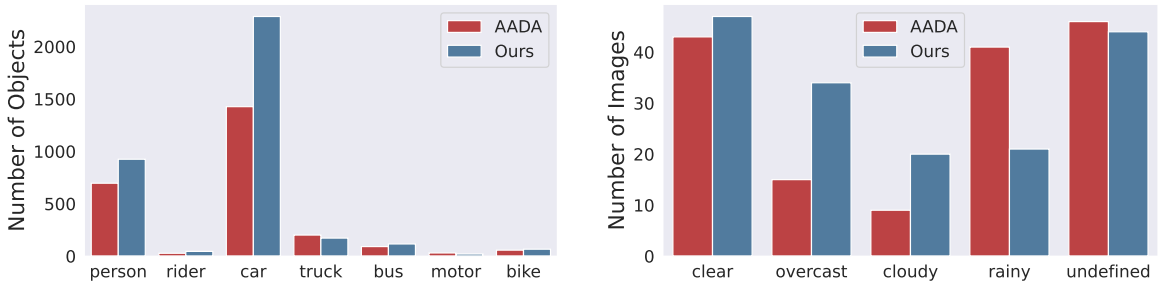
AADA Ours

Figure S1. **Top-10 images sampled from the target domain at round 1 in the Cityscapes → BDD100k scenario.** We compare the images sampled by AADA and our proposed method.

Yang. Progressive domain adaptation for object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 749–757, 2020. 3

[7] Vaishnavi Khindkar, Chetan Arora, Vineeth N Balasubramanian, Anbumani Subramanian, Rohit Saluja, and C.V. Jawahar. To miss-attend is to misalign! residual self-attentive feature alignment for adapting object detectors. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3632–3642, 2022. 3

[8] Xianfeng Li, Weijie Chen, Di Xie, Shicai Yang, Peng Yuan, Shiliang Pu, and Yueting Zhuang. A free lunch for unsupervised domain adaptive object detection without source data. In *Proceedings of the AAAI Conference on Artificial Intelli-*



(a) Number of objects per category

(b) Number of images per weather

Figure S2. **Statistical comparison of sampled objects and attributes in the Cityscapes → BDD100k scenario.** (a) is the sum for each category of objects in all sampled images. (b) is the sum for each weather in all sampled images.

Table S4. Comparison of performance at value of dropout rate η and threshold value γ of pseudo-label with variance

η	0.1	0.2	0.3	0.4	0.5
$\gamma = 0.1$	**63.1**	62.2	62.7	62.6	61.7
γ	0.001	0.01	0.05	0.1	0.2
$\eta = 0.1$	0.9	62.9	61.8	**63.1**	61.8

Table S5. Comparison of GPU hours with and without our method in the **Cityscapes** → **BDD100k** scenario

Process	w/o s^{fn}	w/ s^{fn}
Model Training	68.1	68.1
FNPM Training	—	5.7
Active Sampling	3.8	3.7
Total	71.9	77.5

gence, pages 8474–8481, 2021. 3

[9] Yu-Jhe Li, Xiaoliang Dai, Chih-Yao Ma, Yen-Cheng Liu, Kan Chen, Bichen Wu, Zijian He, Kris Kitani, and Peter Vajda. Cross-domain adaptive teacher for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7581–7590, 2022. 3

[10] Kuniaki Saito, Yoshitaka Ushiku, Tatsuya Harada, and Kate Saenko. Strong-weak distribution alignment for adaptive object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6956–6965, 2019. 3

[11] Jong-Chyi Su, Yi-Hsuan Tsai, Kihyuk Sohn, Buyu Liu, Subhransu Maji, and Manmohan Chandraker. Active adversarial domain adaptation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 739–748, 2020. 3, 4

[12] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2017. 2

[13] Dan Wang and Yi Shang. A new active labeling method for deep learning. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pages 112–119, 2014. 4

[14] Chang-Dong Xu, Xing-Ran Zhao, Xin Jin, and Xiu-Shen Wei. Exploring categorical regularization for domain adaptive object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11724–11733, 2020. 3

[15] Wenzhang Zhou, Dawei Du, Libo Zhang, Tiejian Luo, and Yanjun Wu. Multi-granularity alignment domain adaptation for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9581–9590, 2022. 3

[16] Wenzhang Zhou, Heng Fan, Tiejian Luo, and Libo Zhang. Unsupervised domain adaptive detection with network stability analysis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6986–6995, 2023. 3