Supplementary Material for Masked Retraining Teacher-Student Framework for Domain Adaptive Object Detection

Zijing Zhao[1] Sitong Wei[1] Qingchao Chen[2] Dehui Li[3] Yifan Yang[3] Yuxin Peng[1] Yang Liu[1*]
[1]Wangxuan Institute of Computer Technology, Peking University
[2]National Institute of Health Data Science, Peking University [3]Tencent Intelligent Mobility
zijingzhao@stu.pku.edu.cn {weisitong, qingchao.chen, pengyuxin, yangliu}@pku.edu.cn
{dehuili, lvanyang}@tencent.com

In this additional material, we offer a more comprehensive assessment of our proposed MRT. In Section 1, we investigate the generalization of our proposed selective retraining. In Section 2, we provide details and discuss the rationality and effectiveness of our proposed dynamic threshold. In Section 3 we furnish additional details regarding the implementation of the proposed approach. In Section 4 we present more qualitative examples. In Section 5 we deliberate on the limitations and future directions of our work.

1. Further Discussion on Selective Retraining

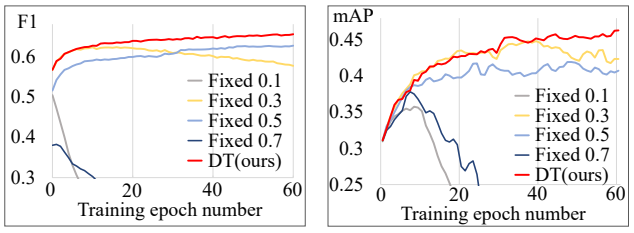
In this section, we apply our proposed selective retraining mechanism to some existing approaches. As shown in Table 1, the selective retraining mechanism is a simple yet effective way to help the student model jump out of local optimums in teacher-student framework.

We evaluated the effectiveness of our proposed selective retraining mechanism on the competitive methods AT [3] which is based on the Faster R-CNN detector with a teacher-student framework. Our results demonstrate that the selective retraining mechanism is also beneficial for Faster R-CNN based detectors, helping them overcome local optima biased to incorrect pseudo labels encountered during teaching process. Additionally, we found that transformer-based detector(our proposed MRT) benefits more from the selective retraining mechanism compared to Faster R-CNN. This is because transformer models are more susceptible to overfitting on incorrect pseudo labels when pretraining data is limited. Specifically, Deformable DETR's encoder and decoder components have a larger number of parameters compared to the detection head of Faster R-CNN. As a result, after re-initialization, the improved decoder component of Deformable DETR facilitates other modules to converge towards a more favorable optimum, whereas the enhanced detection head of Faster R-CNN exhibits only marginal performance gains.

*Corresponding author

Method	mAP
AT[3]	47.4
AT[3]+retrain	47.9
MRT(ours) w/o retrain	48.3
MRT(Ours)	**51.2**

Table 1. **Results of adopting Selective Retraining on different detectors** on *Cityscapes* to *Foggy Cityscapes(0.02)*.



(a) Pseudo label f1 curve (b) Output mAP curve

Figure 1. **(a) Pseudo label f1 and (b) output mAP of different threshold strategies** on *Cityscapes* to *Foggy Cityscapes(0.02)*.

2. Further Discussion on Dynamic Threshold

In this section, we first provide more details of our proposed dynamic threshold (DT), and then discuss its rationality and effectiveness.

We initialize the thresholds for each category by a same value and dynamically update them based on the predicted confidence scores of the source domain instances. Specifically, threshold δ_c for category c is initialized by a shared hyper-parameter δ_0, and is updated every epoch by:

$$\delta_c \leftarrow \gamma \cdot \delta_c + (1-\gamma) \cdot a \cdot (\overline{l_c})^b \tag{1}$$

where $\overline{l_c}$ denotes the mean confidence score of source domain instances (only positive ones which match with ground truths) of category c in the whole dataset. γ, a and b are hyper-parameters. γ decides the influence of predicted confidence scores. $b \in (0, 1)$ provides a convex function

Hyper-parameter	Description	City2Foggy	City2BDD	Sim2City
N_c	Number of categories for classification head	9	9	4
N_l^{enc}	Number of encoder layers	6	6	6
N_l^{dec}	Number of decoder layers	6	6	6
N_l^{aux}	Number of MAE auxiliary decoder layers	2	2	2
N_q^{dec}	Number of queries for decoder	300	300	300
N_q^{aux}	Number of queries for MAE auxiliary decoder	882	882	882
H	Number of hidden dimension for deformable attention	256	256	256
F	Number of feedforward dimension for deformable attention	1024	1024	1024
L	Number of feature levels for deformable attention	4	4	4
M	Number of heads for deformable attention	8	8	8
K	Number of reference points for each attention head	4	4	4
$dropout$	Ratio for dropout in Deformable DETR	0.0	0.0	0.0
B	Batch Size during training	8	8	8
lr	Learning rate for modules except backbone and projection	2×10^{-4}	2×10^{-4}	2×10^{-4}
lr_{bac}	Learning rate for backbone and projection modules	2×10^{-5}	2×10^{-5}	2×10^{-5}
β_{bac}	Coefficient of discrimination loss after backbone $\mathcal{L}_{dis}^{bac}$	0.3	0.3	0.3
β_{enc}	Coefficient of discrimination loss after encoder $\mathcal{L}_{dis}^{enc}$	1.0	1.0	1.0
β_{dec}	Coefficient of discrimination loss after decoder $\mathcal{L}_{dis}^{dec}$	1.0	1.0	1.0
λ_{unsup}	Coefficient of unsupervised loss $\mathcal{L}_{unsup}$	1.0	1.0	1.0
λ_{mask}	Coefficient of MAE loss $\mathcal{L}_{mask}$	1.0	1.0	1.0
α	EMA update ratio	0.9996	0.9996	0.9996
r_{mask}	Mask ratio in MAE branch	0.8	0.8	0.8
γ	Hyper-parameter γ in dynamic threshold	0.9	0.9	0.9
a	Hyper-parameter a in dynamic threshold	0.5	0.9	0.5
b	Hyper-parameter b in dynamic threshold	0.5	0.5	0.5
δ_0	Initialization of thresholds in dynamic threshold	0.3	0.3	0.3
δ_{max}	The upper-bound of thresholds in dynamic threshold	0.45	0.6	0.5
E_{pre}	MAE branch with source data training epoch number	90	100	90
E_{teach}	Teacher-student training epoch number	80	20	20
E_{decay}	After E_{decay} epochs in teaching stage, we drop the MAE branch	10	5	10
E_{reinit}	Re-initialization epoch for selective retraining	40	10	10

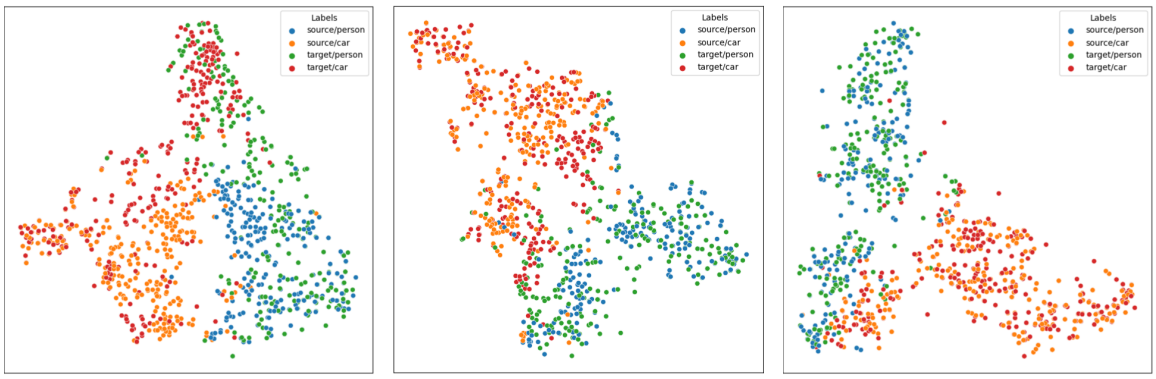
Table 2. **Detailed hyper-parameters for each benchmark.** "City2Foggy" denotes *Cityscapes* to *Foggy Cityscapes(0.02)*, "City2BDD" denotes *Cityscapes* to *BDD100k-daytime*, "Sim2City" denotes *Sim10k* to *Cityscapes(car)*.

and a provides a linear projection, together preventing the threshold from being too high or too low. Moreover, we set a fixed threshold upper-bond δ_{max}, i.e. if any threshold reaches δ_{max}, we no longer update it.

We assume that the threshold selecting pseudo labels for target images is strongly correlated to the predicted confidence scores of source instances. The reason is that confidence scores continue to increase during training, and With a fixed threshold on such scores, the number of selected pseudo boxes increases unboundedly and brings massive error ones. Our DT which increases according to confidence scores helps reduce such error accumulation. Our DT considers the data distribution on categories. However, the ground truth categories of target proposals are unavailable. With source+MAE training stage pulling the distributions of two domains together, we can use source confidence scores per-category as an alternative. Figure 1(a) shows that our DT selects pseudo labels with higher quality (f1 score) compared to fixed thresholds, and thus achieves better performance as shown in Figure 1(b), demonstrating DT's superiority.

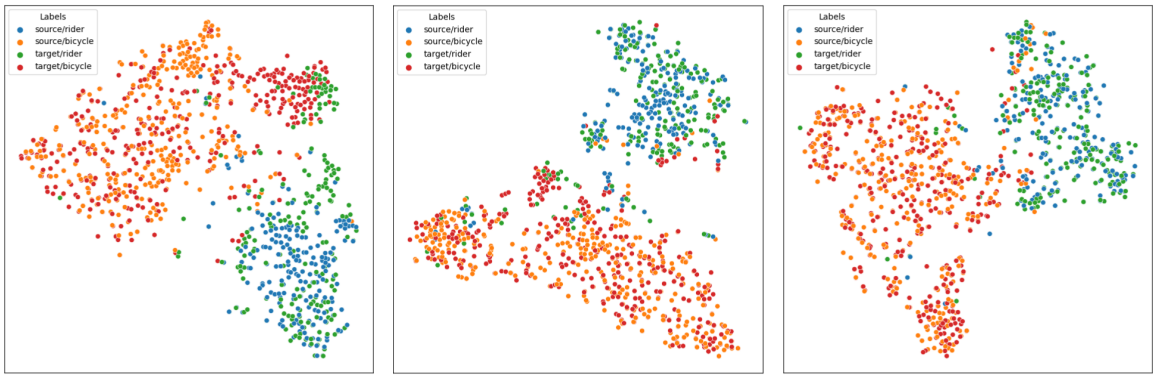
3. Implementation Details

Training stages: For λ_{mask} which controls the influence of MAE branch, in pretraining stage, λ_{mask} is initially 0 and rises later, while in teaching stage, λ_{mask} decays in later stage. As has been discussed in Section 4.3 and Section 5.4 of the main body, hard decay performs slightly better than linear decay, and largely decrease the computation cost. Thus in practice, we split the training into 3 stages: (1) source-only training stage: training with only source labeled data; (2) cross-domain-MAE training stage: training with source data and MAE branch for target data; (3)



(a) Source-only (b) Source+MAE (c) MRT (ours)

Figure 2. **Qualitative ablation: feature visualization of different categories and different domains in *Cityscapes* to *Foggy Cityscapes* by t-SNE.** The color orange, red, blue and green denotes source "car", target "car", source "person" and target "person" respectively.



(a) Source-only (b) Source+MAE (c) MRT (ours)

Figure 3. **Qualitative ablation: feature visualization of different categories with fewer samples and different domains in *Cityscapes* to *Foggy Cityscapes* by t-SNE.** The color orange, red, blue and green denotes source "bicycle", target "bicycle", source "rider" and target "rider" respectively.

MRT teaching stage: training with MAR branch as well as teacher-student framework. Three stages are conducted in sequence. We denote the total training epoch number of stage (1) and (2) as E_{pre}, and the training epoch number of stage (3) as E_{teach} in Table 2.

Explanation of hyper-parameters: Detailed settings for each benchmark is listed in Table 2. We use ImageNet pretrained ResNet-50 as the backbone following [6, 4, 5, 1, 2]. Among the hyper-parameters, for N_c, i.e. number of categories for classification head, we use the number of categories in the source domain dataset which is larger than the evaluated target dataset, following [6, 4]. Since the MAE decoder reconstructs the feature map based on the fixed number of queries, we crop and resize the input images to a fixed size 666×1333, and reconstruct the last layer of feature map with size 21×42 (produced by ResNet-50 backbone). Thus, the query number of MAE auxiliary decoder is $21 \times 42 = 882$. We empirically observe that turning off the dropout in Deformable DETR gets better results. For learning rates, we keep the learning rate of the backbone and the projection modules 10 times smaller than the learning rate of transformer modules following [6]. We build our code on top of the code base of [6] and [4]. Our code is available at https://github.com/JeremyZhao1998/MRT-release.

4. More Qualitative Examples

In this section, we provide more qualitative examples including visualization of pseudo labels and feature distributions of each category.

As a supplement to Figure 4 in the main body which shows the alignment of two domains, we provide features from different categories and different domains in Figure 2 and Figure 3. Figure 2(a) illustrates that the model trained solely on source data can differentiate between various cat-

(a) Source-only (b) Baseline (c) MRT w/o Retrain (d) MRT (ours) (e) Ground truth

Figure 4. **Qualitative ablation: pseudo labels for *Sim10k* to *Cityscapes(car)*.**

(a) Source-only (b) Baseline (c) MRT w/o Retrain (d) MRT (ours) (e) Ground truth

Figure 5. **Qualitative ablation: pseudo labels for *Cityscapes* to *BDD100k-daytime*.**

egories within the source domain (orange and blue points). However, it fails to align the two domains (red and orange points, as well as blue and green points) and struggles to distinguish between different categories in the target domain (red and green points cluster together in some areas). The integration of the MAE branch shown in Figure 2(b) results in a slightly improved ability for the model to differentiate between the two categories in the target domain, with fewer red and green points overlapping. In Figure 2(c), our proposed MRT not only aligns the two domains but also accurately distinguishes between the two categories. Figure 3 shows the same conclusion on categories with fewer samples, indicating that our proposed method is data efficient and works well with long-tail category sample distributions.

As a supplement to Figure 3 in main body, we provide visualization of pseudo labels on *Sim10k* to *Cityscapes(cars)* in Figure 4 and *Cityscapes* to *BDD100k-daytime* in Figure 5. Our proposed MAE branch and selective retraining significantly improves the quality of pseudo labels in multiple adaptation scenarios.

5. Limitations

We want to address the limitations of our approach. Firstly, the use of the teacher-student framework and the MAE branch increases the computational cost. This leads to a larger memory space requirement for GPUs, limiting the training batch size. Moreover, the retraining mechanism further adds to the overall training time. On the bright side, in our proposed approach, both the student branch and the MAE branch is no longer required during inference, which keeps the inference time and space cost equivalent to standard Deformable DETR (our base detector). We will explore ways to improve the training process and increase convergence speed in the future.

References

[1] Kaixiong Gong, Shuang Li, Shugang Li, Rui Zhang, Chi Harold Liu, and Qiang Chen. Improving transferability for domain adaptive detection transformers. *arXiv preprint arXiv:2204.14195*, 2022. 3

[2] Wei-Jie Huang, Yu-Lin Lu, Shih-Yao Lin, Yusheng Xie, and Yen-Yu Lin. Aqt: Adversarial query transformers for domain adaptive object detection. In *31st International Joint Conference on Artificial Intelligence, IJCAI 2022*, pages 972–979. International Joint Conferences on Artificial Intelligence, 2022. 3

[3] 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*, pages 7581–7590, 2022. 1

[4] Wen Wang, Yang Cao, Jing Zhang, Fengxiang He, Zheng-Jun Zha, Yonggang Wen, and Dacheng Tao. Exploring sequence feature alignment for domain adaptive detection transformers. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 1730–1738, 2021. 3

[5] Jinze Yu, Jiaming Liu, Xiaobao Wei, Haoyi Zhou, Yohei Nakata, Denis Gudovskiy, Tomoyuki Okuno, Jianxin Li, Kurt Keutzer, and Shanghang Zhang. Cross-domain object detection with mean-teacher transformer. *arXiv preprint arXiv:2205.01643*, 2022. 3

[6] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*, 2020. 3