Complementary Masking and Cyclic Training for Unsupervised Domain Adaptive Object Detection

Jing Teng, *Senior Member, IEEE*, Zhiwei Zhu, Xujie Long, Mengyang Pu*, Tian Wang, *Senior Member, IEEE*, Ruifeng Shi, *Senior Member, IEEE*, You Lv, Hichem Snoussi, Jonathan Li, *Fellow IEEE*

***Abstract*—Unsupervised Domain Adaptive Object Detection (UDAOD) transfers knowledge from a labeled source domain to an unlabeled target domain, which boosts the adaptability of detectors across diverse data distributions. While advances in the teacher-student framework show significant promise for bridging domain gaps, two challenges remain: the unreliability of the pre-trained model due to target domain ignorance, and the negative impact caused by low-quality pseudo-labels. To address these challenges, we propose a two-stage teacher-student framework, termed Complementary Masking and Cyclic Training (CMCT). Specifically, we introduce a complementary masking strategy in the pre-training stage by minimizing the representation discrepancy between the teacher and student networks for same objects. This enables target domain knowledge learning and improves the reliability of the pre-trained model. In the subsequent self-training stage, we adopt a cyclic training strategy with iterative mixed pseudo-label training and adversarial feature alignment. This strategy alleviates the negative impact of low-quality pseudo-labels while providing more reliable supervision signals. Extensive experiments on three cross-domain scenarios, including Cityscapes→Foggy Cityscapes, Sim10K→Cityscapes, and Cityscapes→BDD100K, demonstrate that the proposed CMCT achieves competitive performance. In particular, the method obtains 53.4% mAP$_{@50}$ in the Cityscapes→Foggy Cityscapes scenario, exceeding the previous performance of 52.5% mAP$_{@50}$.**

***Index Terms*—Unsupervised domain adaptive object detection, teacher-student framework, data augmentation.**

I. Introduction

GENERIC object detection aims to localize and categorize objects in images. As a fundamental task in computer vision, it has a wide variety of applications, such as defect detection [1], object tracking [2], and autonomous driving [3]. However, the assumption that training (source domain) and testing (target domain) data follow the same distribution is not always valid in real-world scenarios. The shift distribution has been shown to cause a decrease in object detection accuracy across different domains [4], [5]. Thus, unsupervised domain adaptive object detection (UDAOD) emerges as a key solution to enable cross-domain generalization and mitigate accuracy degradation due to distribution shifts.

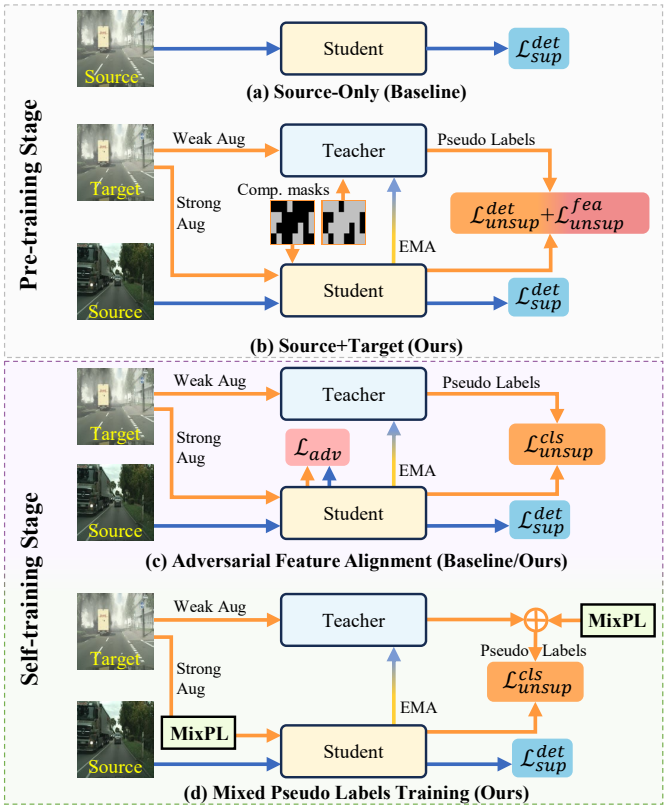


Fig. 1: **Teacher-Student framework comparison**, where (a) and (c) depict the pre-training and self-training stage of the baseline teacher-student framework. (b) shows the proposed pre-training stage, while (c) and (d) illustrate the proposed self-training stage.

UDAOD methods [6]–[10] generalize object detection models trained on a labeled source domain to an unlabeled target domain without additional annotations. Early UDAOD approaches utilize adversarial feature alignment [6]–[8] and image-style translation [9], [10] to reduce discrepancies in both appearance and feature distributions across domains. However, the absence of target domain annotations constrains the performance of these methods.

Recently, the Teacher-Student framework [11] dominates UDAOD, leveraging its capacity to generate pseudo-labels for the target domain. Existing methods [12], [13] typically use a two-stage training pipeline, namely pre-training and self-

This work was supported by the National Natural Science Foundation of China (Grant No. 62373148, 62301220), and the Fundamental Research Funds for the Central Universities (Grant No. 2025MS024, 2025JC005).

Jing Teng, Zhiwei Zhu, Xujie Long, Mengyang Pu, Ruifeng Shi, and You Lv are with the School of Control and Computer Engineering, North China Electric Power University, Beijing 102202, China, are also with Yanzhao Electric Power Laboratory of North China Electric Power University (e-mail:{jing.teng, zhiwei.zhu, longxujie, mengyang.pu, shi.ruifeng, you.lv}@ncepu.edu.cn)

Tian Wang is with the School of Artificial Intelligence, SKLSDE, Beihang University, and Zhongguancun Laboratory, Beijing 100080, China (e-mail: wangtian@buaa.edu.cn)

Hichem Snoussi is with the University of Technology of Troyes, 12 rue Marie Curie, CS 42060, 10004 Troyes CEDEX, France (e-mail: hichem.snoussi@utt.fr).

Jonathan Li is with Geospatial Intelligence and Mapping Lab, Department of Geography and Environmental Management, University of Waterloo, Waterloo, Ontario N2L 2G1, Canada. (e-mail: junli@uwaterloo.ca)

training. Firstly, the student network is pre-trained on labeled source domain data, as illustrated in Fig. 1 (a). Secondly, the teacher network is initialized from the parameters of the pretrained student, then generates pseudo-labels for unlabeled target data and combines them with source labels to supervise the student network, as shown in Fig. 1 (c). Moreover, the student model integrates with Gradient Reversal Layer (GRL)-based [14] adversarial alignment to bridge domain gaps and suppress false positives.

However, due to the lack of target domain knowledge in the pre-training stage, the teacher network that is initialized by the pre-trained student network tends to generate sparse and inaccurate pseudo-labels. These low-quality pseudo-labels may then lead the student network in the self-training stage to acquire erroneous semantic features, thereby hindering the object detection capabilities. Although simple thresholding and GRL-based Adversarial Feature Alignment (AFA) can filter out a subset of noisy pseudo-labels, residual noise persists and propagates back to the teacher network through exponential moving average (EMA) parameter updates, causing the student network to gradually overfit noisy signals and degrade performance.

To tackle these issues, we propose an innovative two-stage teacher-student framework, termed **C**omplementary **M**asking and **C**ycle **T**raining (CMCT), for improving the reliability of the pre-trained network and reducing the negative impact of low-quality pseudo-labels. Specifically, our novel pre-training stage, as shown in Fig. 1 (b), employs the teacher-student framework with a complementary masking strategy instead of the traditional single-student network designs. This pretraining design minimizes the discrepancy in category representations between teacher and student networks for the same objects. The student is encouraged to capture contextual cues of objects from the complementary masked regions, which enables the student to learn more representative target domain features and facilitates the teacher in generating higher-quality pseudo-labels. In addition to AFA, we integrate Mixed Pseudo-Label Training (MPLT) in the self-training stage, as shown by Fig. 1 (c) and (d). Furthermore, instead of naïvely combining AFA and MPLT, a cyclic training strategy that alternates between them is adopted. Concretely, AFA reduces false positives through the alignment of feature distributions between the source and target domains. In parallel, MPLT improves the robustness of the student network by training on mixed target-domain samples. The alternation between AFA and MPLT not only mitigates mutual interference but also leverages their complementary strengths.

Therefore, our contributions can be summarized as follows:

- A novel two-stage teacher-student framework named CMCT is proposed for UDAOD, which effectively tackles the unreliable pre-trained teacher network and the adverse effects of low-quality pseudo-labels on the student network.
- A complementary masking strategy is introduced to improve the quality of pseudo-labels generated by the pre-trained teacher network.
- A cyclic training strategy oscillates between MPLT and AFA, which concurrently diminishes the negative impact of low-quality pseudo-labels while promoting the quantity of correct ones.
- Extensive experiments demonstrate that CMCT produces competitive performance on three widely used cross-domain datasets: Cityscapess→Foggy Cityscapes, Sim10K→Cityscapes, and Cityscapess→BDD100K. Notably, our method achieves a superior performance of 53.4 $\text{mAP}_{@50}$ in the Cityscapes→Foggy Cityscapes.

II. Related Work

A. Unsupervised Domain Adaptive Object Detection

Unsupervised domain adaptive object detection bridges source-target domains by training models on labeled sources, with adversarial feature alignment [6]–[8], [15], image style transfer [9], [10], and self-training [13], [16]–[21]. For example, Chen *et al*. [6] investigates adversarial feature alignment for aligning image-level and instance-level features. Subsequent works [7], [8], [15] apply different aspects of feature alignment with diverse mechanisms. Zhang *et al*. [9] use CycleGAN [22] to learn the mutual mapping functions between the source and target domains, and align their features at both the data distribution and semantic levels. Hsu *et al*. [10] transform target images to source-like images by using an image-image translation module, and then reduce domain gap with the help of adversarial feature alignment.

Recently, self-training methods [13], [18], [20] based on the teacher-student (TS) framework have made significant progress. MTOR [18] extends the TS framework by integrating region-level, inter-graph, and intra-graph consistency constraints to optimize structured relationship modeling. Adaptive Teacher [13] reduces false positives in pseudo-labels by combining weak-strong data augmentations with adversarial feature alignment. Probabilistic Teacher [20] uses uncertainty-guided self-training to jointly improve classification and localization adaptation. Despite these advancements, existing methods still face two major challenges: (1) The reliability of the pre-trained teacher network is limited by target domain knowledge. (2) The adverse effects of low-quality pseudo-labels on the student network in the self-training stage. Thus, we introduce a complementary masking strategy to learn target domain knowledge during the pre-training stage and employ the cyclic training strategy to mitigate the adverse effects of low-quality pseudo-labels in the self-training stage.

B. Model Pre-training

Pre-training strategies such as MAE [23], MoCo [24], and SimCLR [25] are indispensable in transfer learning, where the model is pre-trained on large-scale datasets and then adapts to downstream or cross-domain tasks. In unsupervised domain-adaptive object detection (UDAOD), the teacher network trained on source data often exhibits domain bias and yields low-quality pseudo-labels on the target domain. To mitigate this, recent efforts [21], [26] leverage target domain data for pre-training. Specifically, to enable the student network to effectively capture target domain representations, MRT [21] reconstructs the masked features of the target domain by

introducing an auxiliary decoder. MTM [26] employs image style transfer to synthesize target-like images for pre-training. It further enhances the robustness of the student network through masked domain query and token-level feature alignment. While effective, they tend to focus more on learning image-level contextual representations rather than object-level semantics, which may be suboptimal for object detection. Recently, SeqCo-DETR [27] applies complementary masks to an online network and a momentum network, and promotes the learning of object-level contextual features by enforcing consistency between their sequence outputs.

Inspired by SeqCo-DETR [27], we integrate the complementary masks into the pre-training stage of the TS framework. However, unlike its input-level masking, our feature-level masking could effectively capture object-level semantics and enhance the quality of pseudo-labels of the pre-trained model.

C. Noisy Labeling

Noisy labels pose critical challenges for UDAOD and degrade prediction accuracy for both categories and bounding boxes. The recent Mixup-based approaches [28], [29] show promise in cross-domain scenarios. For instance, IIMT [28] enhances target domain generalization through both cross-domain and intra-domain mixed training. STM [29] builds on adversarial feature alignment and cyclical data distillation, and employs Mixup [30] to generate off-distribution samples for pseudo-label noise dilution. However, scarce pseudo-labels sharply diminish the regularization efficacy of Mixup [30] in existing methods. MixPL [31] extends Mixup with Mosaic augmentation [32] to reduce the adverse effects of noisy pseudo-labels and detect small objects. Nevertheless, applying this approach directly to the cross-domain scenario may yield suboptimal performance due to the domain gap. In this paper, we propose a cyclic training strategy that alternately performs mixed pseudo-label training (MPLT) and adversarial feature alignment (AFA). This scheme not only effectively increases the quantity of correct pseudo-labels but also mitigates the negative impact of incorrect pseudo-labels.

III. Preliminaries

A. Problem Definition

Given a labeled source domain dataset $\mathbf{D}_s = \left\{\left(x_s^i, y_s^i\right)\right\}_{i=1}^{N_s}$ of size N_s, and an unlabeled target domain dataset $\mathbf{D}_t = \left\{\left(x_t^i\right)\right\}_{i=1}^{N_t}$ of size N_t, where $x_{s/t}^i \in \mathbb{R}^{3\times H\times W}$ is the i-th image of the source s or target t domain, and $y_s = \{(b_j, c_j)\}_{j=1}^{N_o}$ denotes the object detection annotations containing bounding boxes $b_j \in \mathbb{R}^4$ and categories $c_j \in \{1, 2, \ldots, C\}$, with N_o being the number of annotated objects in the image and C representing the total number of categories in the dataset. The goal of UDAOD is to adapt a detection model trained on the labeled source domain to the unlabeled target domain.

B. Review of Teacher-Student Framework

The Teacher-Student (TS) framework consists of a teacher network Ψ_{tch} and a student network Ψ_{stu}, with identical architecture. It works in two stages: a pre-training stage and a self-training stage. In the pre-training stage, the student network $\Psi_{\text{stu}}^{\text{pre}}$ is trained on the labeled source dataset $\mathbf{D}_s$ using the supervised detection loss $\mathcal{L}_{\text{sup}}^{\text{det}}$, which is computed as:

$$\mathcal{L}_{\text{sup}}^{\text{det}} = \mathcal{L}_{\text{sup}}^{\text{box}}\left(\mathbf{x}_s, \mathbf{y}_s\right) + \mathcal{L}_{\text{sup}}^{\text{giou}}\left(\mathbf{x}_s, \mathbf{y}_s\right) + \mathcal{L}_{\text{sup}}^{\text{cls}}\left(\mathbf{x}_s, \mathbf{y}_s\right), \quad (1)$$

where $\mathcal{L}_{\text{sup}}^{\text{box}}$, $\mathcal{L}_{\text{sup}}^{\text{giou}}$, and $\mathcal{L}_{\text{sup}}^{\text{cls}}$ represent the Bounding Box regression loss, Generalized Intersection over Union (GIoU) loss, and Classification loss, respectively. The student network $\Psi_{\text{stu}}^{\text{pre}}$ trained by $\mathcal{L}_{\text{sup}}^{\text{det}}$ is then used to initialize the teacher $\Psi_{\text{tch}}^{\text{self}}$ and student network $\Psi_{\text{stu}}^{\text{self}}$ for self-training.

In the self-training stage, the target-domain images undergo two levels of augmentation: a weak augmentation comprising horizontal flipping and random cropping, and a strong augmentation that includes random color jittering, grayscaling, and Gaussian blurring. The weakly augmented target images $\mathbf{x}_{t_w}$ are input into the teacher network $\Psi_{\text{tch}}^{\text{self}}$ to generate predictions, which are filtered by a confidence threshold to obtain pseudo-labels $\tilde{\mathbf{y}}_t$. The student network $\Psi_{\text{stu}}^{\text{self}}$ is then trained on strongly augmented target images $\mathbf{x}_{t_s}$ together with source images $\mathbf{x}_s$, supervised by the pseudo-labels $\tilde{\mathbf{y}}_t$ and the ground-truth labels $\mathbf{y}_s$, respectively. This weak–strong mechanism helps $\Psi_{\text{tch}}^{\text{self}}$ yield more reliable pseudo-labels while encouraging $\Psi_{\text{stu}}^{\text{self}}$ to learn perturbation-invariant representations in training.

The unsupervised loss $\mathcal{L}_{\text{unsup}}$ is calculated on target data using pseudo-labels $\tilde{\mathbf{y}}_t$, and is applied to the classification task [13], [21], defined as:

$$\mathcal{L}_{\text{unsup}} = \mathcal{L}_{\text{unsup}}^{\text{cls}}(\mathbf{x}_t, \tilde{\mathbf{y}}_t). \quad (2)$$

Therefore, the overall loss function $\mathcal{L}_{\text{self}}$ of the self-training stage is composed of $\mathcal{L}_{\text{sup}}^{\text{det}}$ and $\mathcal{L}_{\text{unsup}}$:

$$\mathcal{L}_{\text{self}} = \lambda_{\text{unsup}}\,\mathcal{L}_{\text{unsup}} + \mathcal{L}_{\text{sup}}^{\text{det}}, \quad (3)$$

where λ_{unsup} denotes the weighting factor for $\mathcal{L}_{\text{unsup}}$.

The parameters of the teacher network $\Psi_{\text{tch}}^{\text{self}}$, denoted as Θ_T, are updated via Exponential Moving Average (EMA) of the student network $\Psi_{\text{stu}}^{\text{self}}$ parameters Θ_S, without gradient propagation. The EMA update rule is defined as:

$$\Theta_T \leftarrow \alpha\Theta_T + (1-\alpha)\Theta_S, \quad (4)$$

where $\alpha \in (0, 1)$ is a momentum term controlling the update rate.

IV. Method

A. Method Overview

The proposed CMCT is built upon the two-stage TS framework. In the pre-training stage shown in Fig. 2, complementary masks are applied independently to the teacher $\Psi_{\text{tch}}^{\text{pre}}$ and student $\Psi_{\text{stu}}^{\text{pre}}$. Then, we minimize the discrepancy of two networks in predictions and feature embeddings to optimize $\Psi_{\text{stu}}^{\text{pre}}$. The self-training stage of CMCT includes two alternating procedures: Mixed Pseudo-Label Training (MPLT) and Adversarial Feature Alignment (AFA). To further clarify the design of the proposed CMCT, we detail the two stages in the following sections.

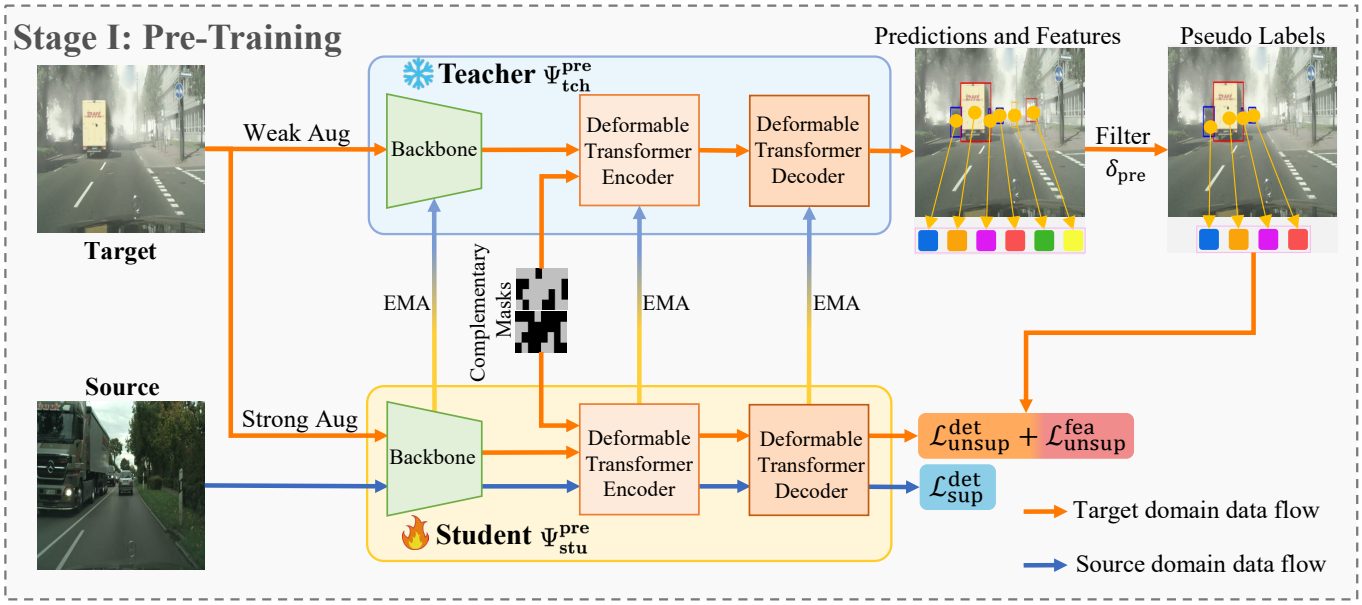


Fig. 2: **Stage I Pre-Training of Our Proposed CMCT**, whose outputs include predicted results and features of the detected objects. The colored squares beneath the result image represent the feature embeddings.

B. Stage I: Pre-training

In the pre-training stage, most existing methods [12], [13], [33], [34] trained only a student network using source-domain data. The trained model is then employed to initialize the teacher network for the subsequent self-training stage. However, with insufficient target domain knowledge, the pre-trained teacher network generates low-quality pseudo-labels, which leads the student network to learn incorrect category representations and misclassify background as foreground. To address this limitation, we incorporate both source and target domain data during the pre-training stage. Furthermore, inspired by [27], we introduce a complementary masking strategy designed to compel the student network Ψ^{pre}_{stu} to acquire discriminative knowledge of objects in the target domain.

The detailed pre-training stage is summarized in Algorithm 1 and illustrated in Fig. 2. Specifically, we instantiate both the teacher Ψ^{pre}_{tch} and student Ψ^{pre}_{stu} using Deformable DETR [35], which includes a ResNet50 backbone, a deformable transformer encoder, and a deformable transformer decoder. We feed weak augmented target images into Ψ^{pre}_{tch}, while source images and strong augmented target images into Ψ^{pre}_{stu}. Since both networks share the same processing pipeline, we use the teacher branch for illustration. The backbone of Ψ^{pre}_{tch} extracts multi-scale features maps denoted as $\mathbf{f}_{tch} = \{f^1_{tch}, f^2_{tch}, ...f^k_{tch}\}$, where $f^k_{tch} \in \mathbb{R}^{C_k \times H_k \times W_k}$ and k represents the feature levels. A random binary mask is then generated with a specified ratio m_{ratio} and a patch size m_{size}. This mask is resized through bilinear interpolation to form a set of multi-scale masks $\mathbf{m}_{tch} = \{m^1_{tch}, m^2_{tch}, ..., m^k_{tch}\}$, where $m^k_{tch} \in \{0,1\}^{H_k \times W_k}$. Thereafter, the multi-scale masks $\mathbf{m}_{tch}$ and the corresponding features $\mathbf{f}_{tch}$ are then fed into the Deformable DETR encoder, where each mask is applied to the value matrix for zero-padding during deformable attention computation. Subsequently, the decoder outputs semantic sequences for generating prediction results $\mathbf{y}_{tch}$. To further obtain the object-level representations and mitigate potential interference, an additional MLP is introduced to project the output sequences into feature embeddings $\mathbf{E}_{tch} \in \mathbb{R}^{N_e \times d_e}$, where N_e denotes the number of embeddings and d_e represents the embedding dimension. Consequently, the teacher network Ψ^{pre}_{tch} produces feature embeddings $\mathbf{E}_{tch}$ along with corresponding predictions $\mathbf{y}_{tch}$. Simultaneously, the student network Ψ^{pre}_{stu} generates feature embeddings $\mathbf{E}_{stu}$ and predictions $\mathbf{y}_{stu}$, following the identical procedure, but with complementary multi-scale masks $\mathbf{m}_{stu} = \overline{\mathbf{m}}_{tch}$.

Thereafter, we set a threshold δ_{pre} to obtain relatively high-quality pseudo-labels $\tilde{\mathbf{y}}_t = \text{Filter}(\mathbf{y}_{tch}, \delta_{pre})$, where each $\tilde{\mathbf{y}}_t$ is associated with a feature embedding $\tilde{\mathbf{E}}_t$. Then, the Hungarian matching [36] is employed to match pseudo-labels $\tilde{\mathbf{y}}_t$ with $\mathbf{y}_{stu}$, generating $\mathbf{y}^{paired}_{stu} = \mathcal{H}(\mathbf{y}_{stu}, \tilde{\mathbf{y}}_t)$. Naturally, the paired embedding $\mathbf{E}^{paired}_{stu}$ is obtained. By minimizing the mean squared error between $\mathbf{E}^{paired}_{stu}$ and $\tilde{\mathbf{E}}_t$, we calculate $\mathcal{L}^{fea}_{unsup}$ as follows:

$$\mathcal{L}^{fea}_{unsup} = \mathcal{L}_{mse}(\mathbf{E}^{paired}_{stu}, \tilde{\mathbf{E}}_t). \tag{5}$$

In addition to the consistency between the paired feature embeddings, the alignment of classification results between Ψ^{pre}_{tch} and Ψ^{pre}_{stu} is also essential for object detection. However, the complementary masking strategy may cause bounding box discrepancies, which prevent Ψ^{pre}_{stu} from integrating the complete object context. To mitigate this, we incorporate additional spatial constraints through the unsupervised box regression loss $\mathcal{L}^{box}_{unsup}$ and Generalized Intersection over Union (GIoU) loss $\mathcal{L}^{giou}_{unsup}$. The overall unsupervised detection loss is formulated as follows:

$$\mathcal{L}^{det}_{unsup} = \mathcal{L}^{box}_{unsup}(\mathbf{x}_t, \tilde{\mathbf{y}}_t) + \mathcal{L}^{giou}_{unsup}(\mathbf{x}_t, \tilde{\mathbf{y}}_t) + \mathcal{L}^{cls}_{unsup}(\mathbf{x}_t, \tilde{\mathbf{y}}_t), \tag{6}$$

where $\tilde{\mathbf{y}}_t$ includes both the classification and bounding box predictions, and $\mathcal{L}^{cls}_{unsup}$ represents the unsupervised classifica-

Algorithm 1 Stage I: Pre-training of CMCT

Input: Labeled source set $\mathbf{D}_s = \{(x_s^i, y_s^i)\}_{i=1}^{N_s}$; unlabeled target set $\mathbf{D}_t = \{x_t^i\}_{i=1}^{N_t}$; student model $\Psi_{\text{stu}}^{\text{pre}}$; teacher model $\Psi_{\text{tch}}^{\text{pre}}$; pre-training epochs T_{pre}; pre-training batch size B_{pre}; filtering threshold δ_{pre}; mask ratio m_{ratio}; mask patch size m_{size}

Output: Pretrained model Ψ^{pre};

for $t_{\text{pre}} = 1$ to T_{pre} **do**
  $N_{\text{iter}} = \min(N_s, N_t)$;
  for $n = 1$ to N_{iter} **do**
    Sample $(\mathbf{x}_s, \mathbf{y}_s) = \{(x_s^i, y_s^i)\}_{i=1}^{B_{\text{pre}}} \sim \mathbf{D}_s$;
    compute $\mathcal{L}_{\text{sup}}^{\text{det}}$ according to Eq. (1);
    Sample $\mathbf{x}_t = \{x_t^i\}_{i=1}^{B_{\text{pre}}} \sim \mathbf{D}_t$;
    Augment $\mathbf{x}_{t_w} = \{x_{t_w}^i\}_{i=1}^{B_{\text{pre}}} = \text{Aug}_w(\{x_t^i\}_{i=1}^{B_{\text{pre}}})$;
    Augment $\mathbf{x}_{t_s} = \{x_{t_s}^i\}_{i=1}^{B_{\text{pre}}} = \text{Aug}_s(\{x_t^i\}_{i=1}^{B_{\text{pre}}})$;
    Generate $\mathbf{f}_{\text{tch}} = \text{Backbone}_{\Psi_{\text{tch}}^{\text{pre}}}(\mathbf{x}_{t_w})$;
    Generate $\mathbf{f}_{\text{stu}} = \text{Backbone}_{\Psi_{\text{stu}}^{\text{pre}}}(\mathbf{x}_{t_s})$;
    Obtain $\mathbf{m}_{\text{tch}} = \text{Mask}(\mathbf{f}_{\text{tch}},\ m_{\text{ratio}},\ m_{\text{size}})$, $\mathbf{m}_{\text{stu}} = \overline{\mathbf{m}}_{\text{tch}}$;
    Generate $(\mathbf{y}_{\text{tch}},\ \mathbf{E}_{\text{tch}}) = \text{Transformer}_{\Psi_{\text{tch}}^{\text{pre}}}(\mathbf{f}_{\text{tch}}, \mathbf{m}_{\text{tch}})$;
    Generate $(\mathbf{y}_{\text{stu}},\ \mathbf{E}_{\text{stu}}) = \text{Transformer}_{\Psi_{\text{stu}}^{\text{pre}}}(\mathbf{f}_{\text{stu}}, \mathbf{m}_{\text{stu}})$;
    Generate pseudo labels $\tilde{\mathbf{y}}_t = \text{Filter}(\mathbf{y}_{\text{tch}},\ \delta_{\text{pre}})$, along with $\tilde{\mathbf{E}}_{\text{t}}$;
    Obtain $\mathbf{y}_{\text{stu}}^{\text{paired}} = \mathcal{H}(\mathbf{y}_{\text{stu}},\ \tilde{\mathbf{y}}_t)$, along with $\mathbf{E}_{\text{stu}}^{\text{paired}}$;
    Compute $\mathcal{L}_{\text{unsup}}^{\text{fea}}$, $\mathcal{L}_{\text{unsup}}^{\text{det}}$, $\mathcal{L}_{\text{unsup}}^{\text{pre}}$, and $\mathcal{L}_{\text{pre}}$ according to Eq. (5)(6)(7)(8);
    New $\Psi_{\text{stu}}^{\text{pre}} = \text{Update}(\Psi_{\text{stu}}^{\text{pre}},\ \mathcal{L}_{\text{pre}})$;
    Update $\Psi_{\text{tch}}^{\text{pre}}$ via EMA according to Eq. (4);
  end for
end for
if $\text{mAP}_{@50}(\Psi_{\text{stu}}^{\text{pre}}) > \text{mAP}_{@50}(\Psi_{\text{tch}}^{\text{pre}})$ **then**
  return $\Psi_{\text{stu}}^{\text{pre}}$;
else
  return $\Psi_{\text{tch}}^{\text{pre}}$;
end if

tion loss. Therefore, the total unsupervised loss of the pre-training combines the feature-level and detection-level losses, given by:

$$\mathcal{L}_{\text{unsup}}^{\text{pre}} = \lambda_{\text{fea}} \mathcal{L}_{\text{unsup}}^{\text{fea}} + \mathcal{L}_{\text{unsup}}^{\text{det}}, \tag{7}$$

where λ_{fea} is a balancing coefficient.

Furthermore, to transfer knowledge from the source domain, the standard supervised detection loss $\mathcal{L}_{\text{sup}}^{\text{det}}$ defined in Eq. (1) is also introduced. Consequently, the overall loss function of the pre-training stage combines supervised and unsupervised components, defined as:

$$\mathcal{L}_{\text{pre}} = \lambda_{\text{unsup}}\, \mathcal{L}_{\text{unsup}}^{\text{pre}} + \mathcal{L}_{\text{sup}}^{\text{det}}, \tag{8}$$

where λ_{unsup} is the weighting factor for the unsupervised loss.

C. Stage II: Self-training

Although our proposed pre-training stage enhances the quality of pseudo-labels, the significant gap between the source and target domains still inevitably leads to sparse and inaccurate pseudo-labels. Prior works [12], [13], [26], [33], [34] typically adopt threshold-based filtering or adversarial feature alignment to correct some of these errors. However, the remaining low-quality pseudo-labels may misguide the student network $\Psi_{\text{stu}}^{\text{self}}$ into learning incorrect category representations, which results in false positives or false negatives. This negative effect is further amplified by the Exponential Moving Average (EMA) mechanism, which accumulates such errors and consequently degrades cross-domain detection performance. Inspired by [31], we propose a cyclic training strategy combining Adversarial Feature Alignment (AFA) and Mixed Pseudo-Label Training (MPLT). The overall workflow is illustrated in details in Fig. 3 and the corresponding pseudocode is presented in Algorithm 2.

1) Adversarial Feature Alignment: AFA aims to correct erroneous pseudo-labels by guiding the student network $\Psi_{\text{stu}}^{\text{self}}$ to bridge the domain gap. Since $\Psi_{\text{stu}}^{\text{self}}$ takes source and target domain images, we apply adversarial loss on $\Psi_{\text{stu}}^{\text{self}}$ to align the features across the two domains. To achieve adversarial training, we treat the backbone, encoder, and decoder as feature extractors F. Each of them is followed by a lightweight CNN-based domain discriminator D to identify the origin domain of features. Specifically, we assign domain labels d = 0 for the source domain and d = 1 for the target domain. The Binary Cross-Entropy loss is employed to train the discriminator:

$$\mathcal{L}_{\text{dis}} = -d \log D(F(x)) - (1-d) \log(1 - D(F(x))). \tag{9}$$

To ensure the feature extractor F produces features that confuse the domain discriminator D, we integrate a Gradient Reversal Layer (GRL) [14] between them for min-max optimization. During backpropagation, GRL reverses the gradient directions to enable adversarial training, with the optimization objective function defined as:

$$\mathcal{L}_{\text{adv}} = \max_{F} \min_{D} \mathcal{L}_{\text{dis}}. \tag{10}$$

This adversarial optimization implicitly performs density ratio estimation [37], where the discriminator aligns the feature distributions of the source and target domains toward unity, thereby promoting domain-invariant representations. Furthermore, we incorporate the standard self-training loss, which includes an unsupervised classification loss $\mathcal{L}_{\text{unsup}}^{\text{cls}}$ and a supervised detection loss $\mathcal{L}_{\text{sup}}^{\text{det}}$. Therefore, the overall loss for AFA is formulated as follows:

$$\mathcal{L}_{\text{AFA}} = \lambda_{\text{unsup}} \mathcal{L}_{\text{unsup}}^{\text{cls}} + \mathcal{L}_{\text{sup}}^{\text{det}} + \lambda_{\text{adv}} \mathcal{L}_{\text{adv}}, \tag{11}$$

where λ_{unsup} and λ_{adv} denote the weighting factor for the unsupervised classification loss and adversarial loss, respectively.

2) Mixed Pseudo-Label Training: As illustrated in Fig. 3, MPLT is built upon the TS framework, enhanced with a MixPL module and a fixed-size target domain cache $\mathbf{C}_t$. The teacher network $\Psi_{\text{tch}}^{\text{self}}$ predictions are first filtered using a high-confidence threshold δ_{self} to obtain reliable pseudo-labels. These pseudo-labels are combined with corresponding

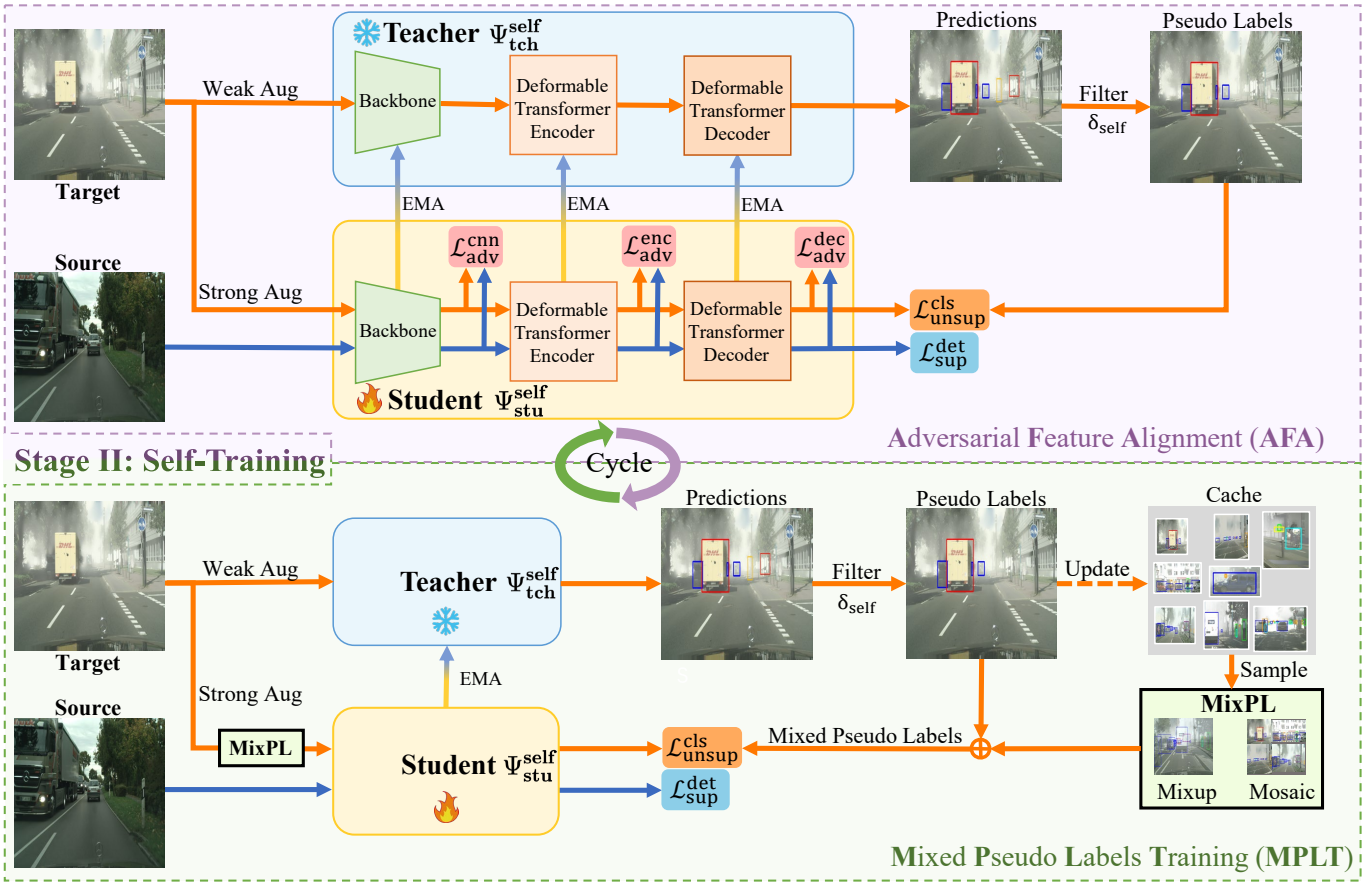


Fig. 3: **Stage II Self-training of Our Proposed CMCT**, where the Adversarial Feature Alignment (AFA) and Mixed Pseudo-Label Training (MPLT) share the same teacher Ψ^{self}_{tch} and student Ψ^{self}_{stu} networks and are trained in a cyclic manner.

strongly augmented target domain images, and then further processed by the MixPL module, which incorporates Mixup and Mosaic augmentations.

Mixup Augmentation. Given the strong augmented target training batch $\{(x^i_{t_s}, \tilde{y}^i_{t_s})\}^{B_{self}}_{i=1}$, where B_{self} denotes the batch size, the pseudo-labels $\{\tilde{y}^i_{t_s}\}^{B_{self}}_{i=1}$ are generated by Ψ^{self}_{tch} using the weakly augmented inputs $\{\tilde{x}^i_{t_w}\}^{B_{self}}_{i=1}$. The MixPL module fetches an equal number of pseudo-labeled samples $\{(x^j_{t_c}, \tilde{y}^j_{t_c})\}^{B_{self}}_{j=1}$ from the target domain cache $\mathbf{C}_t$. The images and pseudo-labels from the cache and training batches are then mixed and concatenated, respectively, using the following equations:

$$\begin{aligned} \hat{x}^k_t &= \sigma x^i_{t_s} + (1-\sigma) x^j_{t_c}, \\ \hat{y}^k_t &= \mathrm{cat}(\tilde{y}^i_{t_s}, \tilde{y}^j_{t_c}), \end{aligned} \tag{12}$$

where $\mathrm{cat}(\cdot)$ represents the concatenation operation, and the mixing coefficient σ is set to 0.5. This Mixup augmentation strategy generates a set of mixed samples $(\hat{\mathbf{x}}_t, \hat{\mathbf{y}}_t) = \{(\hat{x}^k_t, \hat{y}^k_t)\}^{B_{self}}_{k=1}$, where each mixed image is a linear combination of two input batches and the mixed pseudo-labels correspond to the union of their pseudo-labels. These operations can be interpreted as introducing uncertainty-aware interpolation into pseudo-labeled samples, implicitly modeling the uncertainties of feature and label within the target domain. This interpretation is consistent with the principles of uncertainty-aware generative models [38], which aim to explicitly capture predictive uncertainty to enhance robustness under noisy or uncertain supervision. To summarize, Mixup augmentation offers two significant advantages. First, it forces the student network Ψ^{self}_{stu} to learn smoother decision boundaries and reduces its dependence on individual target-domain sample labels, thus exhibiting greater robustness to erroneous pseudo-labels. Second, mixed samples dilute the features of missed detection objects, marking them as hard examples and further reducing their negative impact on the student network Ψ^{self}_{stu} during self-training.

Mosaic Augmentation. The MixPL module samples four pseudo-labeled target pairs $\{(x^i_{t_c}, \tilde{y}^i_{t_c})\}^4_{i=1}$ from the target-domain cache $\mathbf{C}_t$. Each sample is cropped and downsampled, and then placed on four quadrants of a shared canvas. The corresponding pseudo-labels are merged into a new label set $\overline{\mathbf{y}}_t = \{\tilde{y}^i_{t_c}\}^4_{i=1}$, with each quadrant containing one transformed sample to form a composite sample $(\overline{\mathbf{x}}_t, \overline{\mathbf{y}}_t)$. Mosaic augmentation offers two key benefits. First, by combining four images with diverse semantic content, the model is exposed to a broader range of object appearances and backgrounds, which accelerates training. Second, false negatives can be regarded as the result of scale reduction and occlusion transformation of true positives [39]. Therefore, visual features of small objects after Mosaic augmentation exhibit similarity to the features of small-scale and occluded foreground objects in false negatives. After learning such features, Ψ^{self}_{stu} can identify real foreground

Algorithm 2 Stage II: Self-training of CMCT

Input: Labeled source set $\mathbf{D}_s = \{(x_s^i, y_s^i)\}_{i=1}^{N_s}$; unlabeled target set $\mathbf{D}_t = \{x_t^i\}_{i=1}^{N_t}$; pretrained model Ψ_{pre}; student model $\Psi_{\text{stu}}^{\text{self}}$; teacher model $\Psi_{\text{tch}}^{\text{self}}$; target-domain cache $\mathbf{C}_t$; self-training epochs T_{self}; cycle epochs η; AFA epochs β; batch size B_{self}; filtering threshold δ_{self};
Output: Target-domain detector Ψ^{self};
Initialize $\Psi_{\text{tch}}^{\text{self}}$ and $\Psi_{\text{stu}}^{\text{self}}$ with weights from Ψ_{pre};
for $t_{\text{self}} = 1$ to T_{self} **do**
  $N_{\text{iter}} = \min(N_s, N_t)$;
  for $n = 1$ to N_{iter} **do**
    Sample $(\mathbf{x}_s, \mathbf{y}_s) = \{(x_s^i, y_s^i)\}_{i=1}^{B_{\text{self}}} \sim \mathbf{D}_s$;
    compute $\mathcal{L}_{\text{sup}}^{\text{det}}$ according to Eq. (1);
    Sample $\mathbf{x}_t = \{x_t^i\}_{i=1}^{B_{\text{self}}} \sim \mathbf{D}_t$;
    Augment $\mathbf{x}_{t_w} = \{x_{t_w}^i\}_{i=1}^{B_{\text{self}}} = \text{Aug}_w(\{x_t^i\}_{i=1}^{B_{\text{self}}})$;
    Augment $\mathbf{x}_{t_s} = \{x_{t_s}^i\}_{i=1}^{B_{\text{self}}} = \text{Aug}_s(\{x_t^i\}_{i=1}^{B_{\text{self}}})$;
    Generate $\mathbf{y}_{\text{tch}} = \Psi_{\text{tch}}^{\text{self}}(\mathbf{x}_{t_w})$;
    Generate pseudo labels $\tilde{\mathbf{y}}_{t_s}$, $\tilde{\mathbf{y}}_{t_w} = \text{Filter}(\mathbf{y}_{\text{tch}}, \delta_{\text{self}})$;
    if $\mod(t_{\text{self}}, \eta) \leq \beta$ **then**
      Compute $\mathcal{L}_{\text{adv}}$, and $\mathcal{L}_{\text{AFA}}$ according to Eq. (10) (11);
      New $\Psi_{\text{stu}}^{\text{self}} = \text{Update}(\Psi_{\text{stu}}^{\text{self}}, \mathcal{L}_{\text{AFA}})$;
    else
      Sample $\{(x_{t_c}^j, \tilde{y}_{t_c}^j)\}_{j=1}^{B_{\text{self}}+4} \sim \mathbf{C}_t$;
      $(\hat{\mathbf{x}}_t, \hat{\mathbf{y}}_t) = \text{Mixup}\big(\{(x_{t_s}^i, \tilde{y}_{t_s}^i)\}_{i=1}^{B_{\text{self}}}, \{(x_{t_c}^j, \tilde{y}_{t_c}^j)\}_{j=1}^{B_{\text{self}}}\big)$;
      $(\overline{\mathbf{x}}_t, \overline{\mathbf{y}}_t) = \text{Mosaic}\big(\{(x_{t_c}^j, \tilde{y}_{t_c}^j)\}_{j=1}^{4}\big)$;
      Compute $\mathcal{L}_{\text{unsup}}^{\text{MPLT}}$ and $\mathcal{L}_{\text{MPLT}}$ according to Eq. (13) (14);
      New $\Psi_{\text{stu}}^{\text{self}} = \text{Update}(\Psi_{\text{stu}}^{\text{self}}, \mathcal{L}_{\text{MPLT}})$;
    end if
    Update $\Psi_{\text{tch}}^{\text{self}}$ via EMA according to Eq. (4);
  end for
end for
if $\text{mAP}_{@50}(\Psi_{\text{stu}}^{\text{self}}) > \text{mAP}_{@50}(\Psi_{\text{tch}}^{\text{self}})$ **then**
  return $\Psi_{\text{stu}}^{\text{self}}$;
else
  return $\Psi_{\text{tch}}^{\text{self}}$;
end if

objects that were originally misclassified as background and classify them correctly.

After Mixup and Mosaic augmentation, the target domain samples are replaced with the mixed and composite ones in the training batch for student network optimization. We adopt a confidence threshold δ_{self} higher than the one in [31], as we observe that a higher confidence threshold yields better performance, which is demonstrated in our ablation study in Section V-E.

The unsupervised loss used in the MPLT procedure comprises only the classification terms as follows:

$$\mathcal{L}_{\text{unsup}}^{\text{MPLT}} = \mathcal{L}_{\text{unsup}}^{\text{cls}}(\hat{\mathbf{x}}_t, \hat{\mathbf{y}}_t) + \mathcal{L}_{\text{unsup}}^{\text{cls}}(\overline{\mathbf{x}}_t, \overline{\mathbf{y}}_t). \tag{13}$$

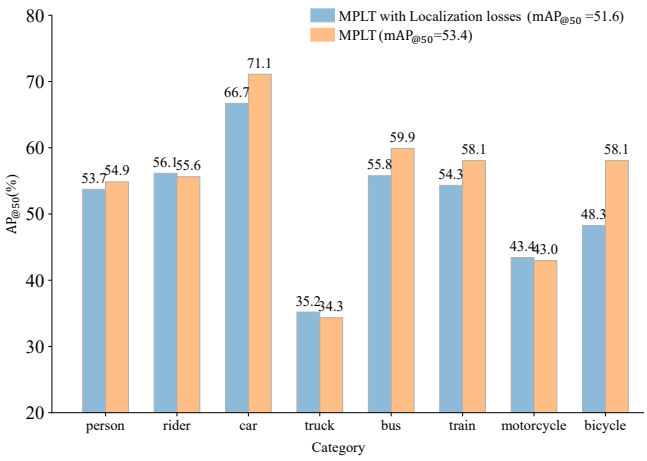


Fig. 4: Visualization of category-wise $\text{AP}_{@50}$ values for MPLT trained with and without the localization loss.

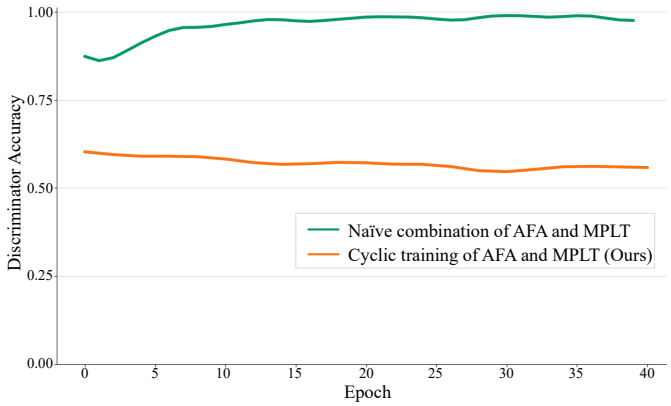


Fig. 5: Comparison of discriminator accuracy on target-domain data between the naïve combination of AFA and MPLT, and the proposed cyclic training.

Since pseudo-label confidence primarily reflects classification certainty rather than localization accuracy [13], [21], [34], incorporating localization loss could compromise the effectiveness of MPLT. As shown in Fig. 4, the inclusion of localization loss in MPLT reduces the overall $\text{mAP}_{@50}$, despite slight category-specific gains in $\text{AP}_{@50}$ for rider, truck, and motorcycle.

Moreover, the supervised detection loss $\mathcal{L}_{\text{sup}}^{\text{det}}$ on labeled source domain data is incorporated to transfer source domain knowledge, contributing to the overall loss:

$$\mathcal{L}_{\text{MPLT}} = \lambda_{\text{unsup}} \mathcal{L}_{\text{unsup}}^{\text{MPLT}} + \mathcal{L}_{\text{sup}}^{\text{det}}, \tag{14}$$

where λ_{unsup} is the weighting factor for the unsupervised loss. After the MPLT procedure is completed, the pseudo-labels $\tilde{\mathbf{y}}_t$ generated by $\Psi_{\text{tch}}^{\text{self}}$ and their corresponding strong augmented target images $\mathbf{x}_{t_s}$ are updated in the cache.

3) Cyclic Training: As aforementioned, AFA and MPLT serve complementary yet potentially conflicting roles. AFA minimizes domain discrepancy to learn domain-invariant features via adversarial loss, while MPLT regularizes training against noisy pseudo-labels through sample mixing. Simultaneous optimization of these conflict objectives leads to mutual interference [52], where mixed features introduced by MPLT

TABLE I: Comparison Results on Cityscapes→Foggy Cityscapes cross-domain scenario, where the best results are shown in bold, and the second-best results are underlined.

Method	Pub'Year	Framework	person	rider	car	truck	bus	train	motor	bicycle	mAP$_{@50}$
SFA [40]	MM'21	Student-Only	46.5	48.6	62.6	25.1	46.2	29.4	28.3	44.0	41.3
SCAN++ [41]	TMM'22		44.2	43.9	57.9	28.2	48.1	51.2	30.1	39.5	42.8
DA-DETR [42]	CVPR'23		49.9	50.0	63.1	24.0	45.8	37.5	31.6	46.3	43.5
BiADT [43]	ICCV'23		50.7	56.3	67.1	28.8	53.7	49.5	38.8	50.1	49.4
GHTCLT [44]	TIV'24		37.5	46.8	54.8	29.3	51.8	46.5	34.1	40.5	42.7
SSTA [45]	TMM'24		50.5	53.0	67.2	24.7	47.7	33.0	36.7	46.6	44.9
CRADA [46]	TMM'24		45.6	43.8	61.8	30.7	51.0	52.1	34.2	38.9	44.8
VLDadaptor [47]	TMM'24		44.7	43.9	62.0	33.8	49.8	55.7	34.6	39.9	45.6
GIBS [48]	ICASSP'25		36.5	47.6	51.9	29.9	50.5	46.5	34.9	37.7	41.9
FGPro [49]	TCYB'25		46.2	54.8	65.6	38.9	63.1	47.9	44.7	50.4	51.4
MTOR [18]	CVPR'19	Teacher-Student	30.6	41.4	44.0	21.9	43.4	40.2	31.7	33.2	35.1
UMT [19]	CVPR'21		33.0	46.7	48.6	34.1	56.5	46.8	30.4	37.3	41.7
MTTrans [12]	ECCV'22		47.7	49.9	65.2	25.8	45.9	33.8	32.6	46.5	43.4
AT [13]	CVPR'22		45.3	55.7	63.6	36.8	64.9	34.9	42.1	51.3	49.3
CMT [33]	CVPR'23		45.9	55.7	63.7	39.6	**66.0**	38.8	41.4	51.2	50.3
MRT [21]	ICCV'23		52.8	51.7	68.7	35.9	58.1	54.5	41.0	47.1	51.2
MTM [26]	AAAI'24		51.0	53.4	67.2	37.2	54.4	41.6	38.4	47.7	48.9
REACT [50]	TIP'24		52.1	**57.1**	66.3	35.0	56.7	52.8	42.9	53.8	52.1
CAT [34]	CVPR'24		44.6	**57.1**	63.7	**40.8**	**66.0**	49.7	**44.9**	53.0	52.5
CDMT [51]	TCSVT'25		45.6	56.4	63.8	37.3	64.2	50.5	41.9	**55.7**	51.9
CMCT (Ours)	-		**54.8**	55.6	**71.1**	34.3	59.9	**58.0**	42.9	50.8	**53.4**

could destabilize the discriminator training in AFA, and vice versa. This phenomenon is empirically validated in Fig. 5. The green curve shows that the discriminator accuracy of AFA quickly approaches 1 under the naïve combination of AFA and MPLT. This indicates that the discriminator easily distinguishes source and target domains, thereby invalidating the adversarial optimization in Eq. (10) and hindering the learning of domain-invariant features. To reconcile the competing objectives of AFA and MPLT while preserving their complementary benefits, we propose the Cyclic Training (CT) strategy, grounded in the established principle of balancing competing optimization objectives [53], [54]. CT alleviates this conflict by alternately optimizing MPLT and AFA rather than training them simultaneously, allowing each procedure to converge independently. This rationale is consistent with alternating optimization method [55] that stabilizes convergence in multi-objective settings. Specifically, we first run MPLT for T_{MPLT} epochs, then initialize both $\Psi_{\text{tch}}^{\text{self}}$ and $\Psi_{\text{stu}}^{\text{self}}$ with the best-performing model to boost the robustness of the TS framework against noisy pseudo-labels. The total number of CT epochs is defined as T_{self}, and is partitioned into T_{self}/η cycles, where η denotes the number of epochs per cycle. Within each cycle, MPLT and AFA are alternated to reduce the negative impact of noisy pseudo-labels and increase the quantity of correct pseudo-labels. Each cycle consists of β epochs of AFA, and followed by $\eta-\beta$ epochs of MPLT. The loss of cyclic training is expressed as follows:

$$\mathcal{L}_{\text{cycle}} = \begin{cases} \mathcal{L}_{\text{AFA}} & \text{if} \quad \text{mod}\ (t_{\text{self}},\ \eta) \leq \beta, \\ \mathcal{L}_{\text{MPLT}} & \text{otherwise}, \end{cases} \tag{15}$$

where t_{self} indicates the current number of training epochs. Additionally, to reduce parameters, $\Psi_{\text{tch}}^{\text{self}}$ and $\Psi_{\text{stu}}^{\text{self}}$ are shared across both MPLT and AFA. The CT strategy effectively leverages the complementary advantages of MPLT and AFA, while mitigating potential interference between them. As illustrated by the orange curve in Fig. 5, the discriminator accuracy of AFA remains stable and close to 0.5, indicating the domain-invariant representations of the source and target domains.

V. Experiment

A. Datasets and Evaluation

1) Cityscapes→Foggy Cityscapes: This adaptation setting aims to explore the weather-induced domain gap. The Cityscapes [56] dataset contains street scenes collected from 50 different cities under clear weather conditions, comprising 2,975 training images and 500 validation images. It covers eight core categories: *person*, *rider*, *car*, *truck*, *bus*, *train*, *motor*, and *bike*. In contrast, the Foggy Cityscapes [57] dataset is a synthetic counterpart generated by applying simulated fog effects with varying densities (0.005, 0.01, 0.02) to the original Cityscapes images. Both datasets share identical training and validation splits. We use Foggy Cityscapes (0.02) as our target dataset in this domain adaptation scenario.

2) Cityscapes→BDD100K: This scenario examines the domain gap caused by different camera sensors. BDD100K [58] is a large-scale benchmark dataset for autonomous driving that includes a wide range of weather conditions and diverse urban scenes. In this study, we use the daytime subset, which contains 36,728 training images and 5,258 test images. For evaluation, we focus on seven core categories consistent with the Cityscapes [56] dataset, excluding the *train* category.

3) Sim10k→Cityscapes: This setting investigates the domain gap between synthetic and real-world images. Sim10k [59] is a synthetic dataset generated using the GTA5 game engine, comprising 10,000 images with 58,701 annotated instances, all belonging to the *car* category.

B. Implementation Details

We implement the proposed CMCT using PyTorch, where the Adam optimizer with a base learning rate of $2e-4$

TABLE II: Comparison Result of Cityscapes→BDD100k-daytime, where the best results are shown in bold, and the second-best results are underlined.

Method	Pub'Year	Framework	person	rider	car	truck	bus	mycle	bicycle	mAP$_{@50}$
Source-only [35]	Arxiv'20	Student-Only	38.9	26.7	55.2	15.7	19.7	10.8	16.2	26.2
SFA [40]	MM'21		40.4	27.6	57.5	19.1	23.4	15.4	19.2	28.9
O2Net [60]	MM'22		40.4	31.2	58.6	20.4	25.0	14.9	22.7	30.5
SIGMA [61]	CVPR'22		46.9	29.6	64.1	20.2	23.6	17.9	26.3	32.7
SIGMA++ [62]	TPAMI'23		47.5	30.4	65.6	21.1	26.3	17.8	27.1	33.7
BiADT [43]	ICCV'23		42.0	34.5	59.9	17.2	19.2	17.8	24.4	32.7
Blenda [63]	ICASSP'24		44.5	36.3	64.1	20.0	18.1	24.6	26.9	33.5
MTTrans [12]	ECCV'22	Teacher-Student	44.1	30.1	61.5	25.1	26.9	17.7	23.0	32.6
PT [20]	ICML'22		40.5	39.9	52.7	**25.8**	**33.8**	23.0	28.8	34.9
MRT [21]	ICCV'23		48.4	30.9	63.7	24.7	25.5	20.2	22.6	33.7
MTM [26]	AAAI'24		**53.7**	35.1	**68.8**	23.0	28.8	23.8	28.0	37.3
REACT [50]	TIP'24		-	-	-	-	-	-	-	35.8
CMCT (Ours)	-		51.7	**40.0**	65.4	24.1	21.7	**27.9**	**31.5**	**37.5**

TABLE III: Results of Sim10k→Cityscapes (car), where the best results are shown in bold, and the second-best results are underlined.

Method	Pub'Year	Framework	mAP$_{@50}$
Source-only [35]	Arxiv'20	Student-Only	47.4
O2Net [60]	MM'22		54.1
BiADT [43]	ICCV'23		55.8
DA-DETR [42]	CVPR'23		54.7
CRADA [46]	TMM'24		57.6
GIBS [48]	ICASSP'25		45.3
MTTrans [12]	ECCV'22	Teacher-Student	57.9
MRT [21]	ICCV'23		62.0
MTM [26]	AAAI'24		58.1
REACT [50]	TIP'24		58.6
DeSimPL [64]	TITS'25		55.3
CMCT (Ours)	-		**62.2**

is employed. The weight of the unsupervised loss is set to $\lambda_{\text{unsup}} = 1.0$, and the EMA update weight $\alpha = 0.9996$. In the pre-training stage, the parameters are set as follows, batch size $B_{\text{pre}} = 4$, feature coefficient $\lambda_{\text{fea}} = 2.0$, $m_{\text{ratio}} = 0.2$ and $m_{\text{size}} = 16$ for generating the base mask, $\delta_{\text{pre}} = 0.3$ for pseudo-label filtering, the number of epochs for pre-training $T_{\text{pre}} = 80$. In the self-training stage, the filtering threshold $\delta_{\text{self}} = 0.5$, batch size $B_{\text{self}} = 8$, $\lambda_{\text{adv}} = 1.0$ in AFA, the cache size of MPLT is equal to batch size B_{self}. The MPLT epoch $T_{\text{MPLT}} = 120$, the CT epoch $T_{\text{self}} = 40$, the epochs per cycle $\eta = 3$, and the AFA epochs per cycle $\beta = 2$. For all other hyperparameters, we follow the default settings in prior works [21], [31]. All experiments are conducted on a A100 GPU with 40 GB memory.

C. Comparison with State-of-the-art methods

1) Cityscapes→Foggy Cityscapes: We compare our model with student-only methods including SFA [40], SCAN++ [41], DA-DETR [42], BiADT [43], GHTCLT [44], SSTA [45], CRADA [46], GIBS [48], FGPro [49] and VLDadaptor [47], and teacher-student (TS) methods including MTOR [18], UMT [19], MTTrans [12], AT [13], CMT [33], MRT [21], MTM [26], REACT [50], CAT [34] and CDMT [51]. The results of all methods are taken from their publications.

Quantitative comparison results are shown in Table I. As can be seen, our method CMCT achieves the best performance of 53.4% mAP$_{@50}$ among all compared methods. It outperforms not only the recent baselines like GIBS [48] of 41.9% mAP$_{@50}$ and FGPro [49] of 51.4% mAP$_{@50}$ but also the SOTA Teacher-Student framework methods such as CDMT [51] of 51.9% mAP$_{@50}$ and CAT [34] of 52.5% mAP$_{@50}$, demonstrating superior overall cross-domain detection capability. Moreover, our method leads in object categories such as *person* with 54.8% AP$_{@50}$, *car* with 71.1% AP$_{@50}$ and *train* with 58.0% AP$_{@50}$, and maintains top-tier performance in other categories, showcasing robust detection across multiple object types. The results validates the effectiveness of CMCT in leveraging the teacher-student paradigm to overcome domain shift and pseudo-label noise.

2) Cityscapes→BDD100K: We compare our method with SOTA methods including Source-only [35], SFA [40], O2Net [60], SIGMA [61], SIGMA++ [62], BiADT [43], Blenda [63], MTTrans [12], PT [20], MRT [21], MTM [26], REACT [50]. As shown in Table II, our proposed CMCT achieves the optimal mAP$_{@50}$ of 37.5% among all compared methods. It outperforms the best Student-Only method Blenda [63] of 33.5% mAP$_{@50}$ and SOTA Teacher-Student methods like REACT [50] of 35.8% mAP$_{@50}$ and MTM [26] of 37.3% mAP$_{@50}$, demonstrating superior cross-domain detection capability. Furthermore, CMCT excels across object categories of *rider* with 40.0% AP$_{@50}$, *mycle* with 27.9% AP$_{@50}$ and *bicycle* with 31.5% AP$_{@50}$, showing robust detection for different camera sensor.

3) Sim10k→Cityscapes (car): Table III shows the comparison between our CMCT and competitive SOTA methods, including Source-only [35], O2Net [60], BiADT [43], DA-DETR [42], CRADA [46], GIBS [48], MTTans [12], MRT [21], MTM [26], and REACT [50], DeSimPL [64]. It can be summarized from Table III that Teacher-Student frameworks generally outperforms the student-only ones. Within the TS framework category, CMCT surpasses all peer methods with 62.2% mAP$_{@50}$. CMCT outperforms the second-best MRT [21] by 0.2% mAP$_{@50}$ and significantly outpaces other counterparts like MTM [26] 58.1% mAP$_{@50}$. Even when compared to the most recent methods of GIBS [48] with 45.3% mAP$_{@50}$ and DeSimPL [64] with 55.3% mAP$_{@50}$, CMCT

TABLE IV: Comparison in terms of training iterations, training time, parameters, and inference speed in the Cityscapes→Foggy Cityscapes scenario.

Method	Detector	Iteration	Train Hour	Params	FPS	$mAP_{@50}$
AT [13]	FR CNN	100k	26.3	**40.1M**	24	49.3
CAT [34]	FR CNN	**50k**	**11.7**	45.5M	**25**	52.5
MRT [21]	Def DETR	75k	31.6	53.3M	13.5	51.2
Ours	Def DETR	118k	53.9	50.4M	13.5	**53.4**

TABLE V: Ablation study in the Cityscapes→Foggy Cityscapes scenario, where 'Gain' denotes the performance improvement over the Source-only baseline, and '-' represents that the model fails to converge.

Stage I: Pre-training		Stage II: Self-training			$mAP_{@50}$	Gain
Source-only	CM	AFA	MPLT	CT		
✓					28.9	+0.0
✓		✓			45.1	+16.2
✓			✓		47.8	+18.9
✓		✓	✓		-	-
✓		✓	✓	✓	50.4	+21.5
✓	✓				38.8	+9.9
✓	✓	✓			48.4	+19.5
✓	✓		✓		50.3	+21.4
✓	✓	✓	✓		-	-
✓	✓	✓	✓	✓	**53.4**	+24.5

demonstrates a substantial performance gap, establishing it as the best-performing approach in this scenario.

D. Comparison of Computational Complexity

Table IV compares the proposed CMCT with AT [13], CAT [34], and MRT [21], in terms of the computational complexity and training cost. The architectural distinctions of these representative TS-based methods lie in their detectors. DETR-based methods typically involve larger parameter counts and relatively lower inference speed compared to CNN-based counterparts, but are superior in detection precision owing to the transformer-based architecture and end-to-end learning mechanism. Specifically, AT and CAT adopt FR CNN, exhibiting fewer parameters of 40.1M and 45.5M, respectively, and higher FPS of 24 and 25. In contrast, MRT and our method utilize Def DETR, resulting in larger parameter counts of 53.3M and 50.4M, respectively, with inference speeds of approximately 13.5 FPS. Despite the longer training time and slightly higher computational overhead, our CMCT method achieves the highest $mAP_{@50}$ in the Cityscapes→Foggy Cityscapes adaptation scenario, demonstrating its superior trade-off between accuracy and efficiency.

E. Ablation study

In this section, we conduct ablation studies to validate the effectiveness of each component in our proposed framework. The results are summarized in Table V. All experiments are performed in the Cityscapes→Foggy Cityscapes cross-domain scenario.

TABLE VI: Ablation study of different masking strategies in the Cityscapes→Foggy Cityscapes scenario.

Mask type	No Mask	Random Mask	Single Mask	**CM**
$mAP_{@50}$	51.2	51.8	52.1	**53.4**

1) Complementary Masking (CM): We first evaluate the effectiveness of CM strategy. As shown in Table V, with the introduction of CM during the pre-training stage, our method achieves a notable 9.9% $mAP_{@50}$ improvement, rising from the source-only baseline of 28.9% to 38.8%. This indicates the value of CM in early-stage target domain feature learning. Moreover, initialization with CM yields an additional 3.3% $mAP_{@50}$ gain from 45.1% to 48.4% over the sole adoption of AFA. Similarly, the synergy of CM and MPLT exhibits a 2.5% $mAP_{@50}$ enhancement, as the metric increases from 47.8% to 50.3%. Overall, CM further promotes 3.0% $mAP_{@50}$ against the model pre-trained on the source domain of $mAP_{@50}$ 50.4%. These results coherently underscore the strong initialization of CM for the subsequent self-training stage.

To further validate the necessity of the proposed CM strategy, three masking strategies were designed for comparison: (1) No Mask, where no mask is applied to either $\Psi_{\text{tch}}^{\text{pre}}$ or $\Psi_{\text{stu}}^{\text{pre}}$; (2) Random Mask, where two independently generated random masks are applied to $\Psi_{\text{tch}}^{\text{pre}}$ and $\Psi_{\text{stu}}^{\text{pre}}$, respectively, using the same masking ratio m_{ratio} and patch size m_{size} as CM; and (3) Single Mask, where the mask is applied only to $\Psi_{\text{stu}}^{\text{pre}}$, with identical m_{ratio} and m_{size} as in CM. The results are summarized in Table VI. Without any mask, the model obtained the lowest performance of 51.2% $mAP_{@50}$. Applying a single mask to $\Psi_{\text{stu}}^{\text{pre}}$ outperformed the random masking scheme by 0.3% $mAP_{@50}$, likely due to the interference introduced by uncorrelated random masks during the bipartite matching process. In contrast, our proposed CM achieves the best performance and surpasses the second-best one by 1.3% $mAP_{@50}$, demonstrating both its necessity and superiority.

We further investigate the impact of different m_{size} and m_{ratio} on CM strategy. The results are reported in Table VII. When fixing $m_{\text{size}} = 16$ and varying m_{ratio}, the optimal performance of 53.4% $mAP_{@50}$ was achieved at $m_{\text{ratio}} = 0.2$. A smaller ratio $m_{\text{ratio}} = 0.1$ yields the lowest performance of 51.1% $mAP_{@50}$, as $\Psi_{\text{stu}}^{\text{pre}}$ was unable to capture complete object structures from the limited visible regions. Conversely, larger ratios of 0.3 and 0.4 lead to suboptimal performances of 51.6% and 51.2% $mAP_{@50}$, respectively, likely because of insufficient information extracted by $\Psi_{\text{tch}}^{\text{pre}}$, resulting in noisy pseudo-labels that degrade mutual learning between $\Psi_{\text{tch}}^{\text{pre}}$ and $\Psi_{\text{stu}}^{\text{pre}}$. Next, we fix $m_{\text{ratio}} = 0.2$ and vary m_{size}. Larger patch size of $m_{\text{size}} = 32$ reduce performance to 51.8% $mAP_{@50}$, as small-object information may reside exclusively in either $\Psi_{\text{tch}}^{\text{pre}}$ or $\Psi_{\text{stu}}^{\text{pre}}$, which causes noisy supervision or incomplete feature learning. Conversely, when $m_{\text{size}} = 8$, the model achieves 52.8% $mAP_{@50}$ but still lower than the optimal one of 53.4%. The smaller patch size may encourage $\Psi_{\text{stu}}^{\text{pre}}$ to focus excessively on local details, impeding global semantic learning. Based on these analyses, we adopt $m_{\text{size}} = 16$ and $m_{\text{ratio}} = 0.2$ as the default configuration for all subsequent

TABLE VII: Sensitivity study of Mask Patch Size m_{size} and Mask Ratio m_{ratio} for CM Strategy in the Cityscapes→Foggy Cityscapes scenario.

Mask Patch	Mask Ratio	mAP$_{@50}$
m_{size} = 16	m_{ratio} = 0.1	51.1
	m_{ratio} = 0.2	**53.4**
	m_{ratio} = 0.3	51.6
	m_{ratio} = 0.4	51.2
m_{size} = 8	m_{ratio} = 0.2	52.8
m_{size} = 32		51.8

TABLE VIII: Ablation study of MixPL module in the Cityscapes→Foggy Cityscapes cross-domain scenario.

Mixup	Mosaic	δ_{self}	mAP$_{@50}$
✓	✓	0.3	49.0
✓	✓	0.5	**53.4**
✓	✓	0.7	49.1
✓		0.5	51.4
	✓	0.5	47.7

TABLE IX: Comparison of throughput, training time per epoch, and memory usage with/without MixPL cache in the Cityscapes→Foggy Cityscapes scenario.

Configuration	Throughput	Time/epoch	Memory
W MixPL cache	3.51 iter/s	0.235 h	37946 MiB
W/o MixPL cache	4.09 iter/s	0.202 h	36674 MiB

experiments.

2) Adversarial Feature Alignment (AFA): The results in Table V further confirm the effectiveness of AFA in rectifying unreliable pseudo-labels. The model achieved mAP$_{@50}$ improvements of 16.2% and 19.5% over the non-AFA counterparts, respectively. When removing AFA from the proposed CMCT, the performance decreased by 3.1% mAP$_{@50}$. This indicates that undant imprecise pseudo-labels left $\Psi_{\text{stu}}^{\text{self}}$ with insufficient target domain knowledge, thereby degrading model performance.

3) Mixed Pseudo-Label Training (MPLT): The overall contribution of MPLT could be evaluated by the ablation results presented in Table V. As illustrated, the MPLT module evidently improve the detection performance with mAP$_{@50}$ gains of 18.9% and 21.4% over the non-MPLT counterparts, respectively. Upon the removal of MPLT from the proposed CMCT, the performance drops by 5% in mAP$_{@50}$. These results confirm that MPLT effectively reduce the negative impact of incorrect pseudo-labels and strengthens feature learning robustness, thereby consistently advancing cross-domain object detection performance.

We further investigate the impact of different confidence thresholds for MPLT. When the filtering threshold δ_{self} is set to 0.3, the performance drops sharply by 4.4% mAP$_{@50}$. Increasing δ_{self} to 0.7 yields a suboptimal result of 49.1%, compared with the optimal performance of 53.4% mAP$_{@50}$ obtained at $\delta_{\text{self}} = 0.5$. To maintain consistency in pseudo-label selection, we fix $\delta_{\text{self}} = 0.5$ for both AFA and MPLT.

TABLE X: Sensitivity study of cyclic epochs and ratios in Cyclic Training on Cityscapes→Foggy Cityscapes scenario.

η	2	3	4		5	
AFA:MPLT	1:1	2:1	2:2	3:1	3:2	4:1
mAP$_{@50}$	52.2	**53.4**	52.5	52.5	52.6	52.9

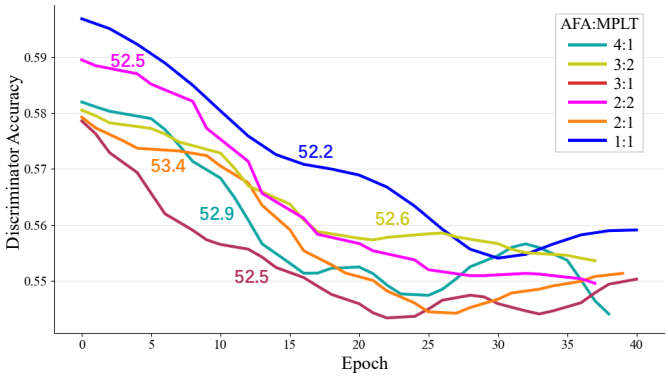


Fig. 6: Backbone discriminator accuracy on the target domain under varying cyclic ratios and cycle epochs in the Cityscapes→Foggy Cityscapes scenario. The colored numbers indicate the mAP$_{@50}$ of each configuration.

Moreover, Table VIII also reports the impact of Mixup and Mosaic augmentation in MPLT. Individual usage of Mixup or Mosaic, mAP$_{@50}$ achieves 51.4% and 47.7%, respectively. Their combination obtains 53.4% mAP$_{@50}$, indicating their complementary impact.

To evaluate the computational overhead introduced by the MixPL cache, we compare throughput (iter/s), per-epoch training time (h), and peak memory usage (MiB) in Table IX. Introducing the cache reduces throughput by 0.58 iter/s, increases per-epoch training time by 0.033 h, and raises peak memory usage by 1,272 MiB. These results indicate that the MixPL cache incurs only minimal computational overhead, with negligible impact on overall training efficiency.

4) Cyclic Training Strategy (CT): As demonstrated in Table V, the removal of CT from CMCT, together with the naïve combination of AFA and MPLT, leads to the collapse of the domain discriminator in AFA shown in Fig. 5. In contrast, cyclic training of AFA and MPLT prevents their mutual interference, thus achieving the optimal performance of 53.4% mAP$_{@50}$.

We further investigate the effects of the cycle epoch η and cyclic AFA:MPLT ratio of $\beta : (\eta - \beta)$ for CT, and visualize corresponding backbone discriminator accuracy. As summarized in Table X and Fig. 6, when the ratios are set to 1:1 and 2:2, the discriminator accuracy remains consistently high throughout training. This indicates that the feature alignment becomes trapped in a local optimum. Consequently, these configurations yield the lowest performances of 52.2% and 52.5% mAP$_{@50}$, respectively. When the ratio is adjusted to 3:2, the discriminator accuracy decreases markedly during the first 20 epochs and then stabilizes around 0.56, resulting in a modest performance improvement to 52.6% mAP$_{@50}$. For higher ratios of 3:1 and 4:1, the discriminator accuracies fluctuate

(a) Source-Only (b) MRT (c) CMCT (Ours) (d) Ground Truth

Fig. 7: Qualitative comparison of Source-Only [35], MRT [21], our proposed CMCT, and Ground Truth in the Cityscapes→Foggy Cityscapes.

TABLE XI: Ablation study of shared student network for cyclic training in the Cityscapes→Foggy Cityscapes scenario.

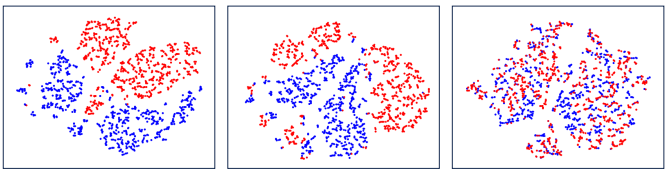
Configuration	Params	$mAP_{@50}$
w/o shared	90.6 M	53.1
Ours	**50.4 M**	**53.4**

considerably. These fluctuations reflect unstable adversarial alignment. Moreover, a larger proportion of AFA weakens MPLT, causing $\Psi_{\text{stu}}^{\text{self}}$ to preserve noisy representations during the prolonged AFA procedure, which leads to suboptimal outcomes. Based on these observations, we set $\eta = 3$ and $\beta = 2$, yielding the cyclic training ratio of AFA:MPLT of $\beta : (\eta - \beta) = 2 : 1$ for all subsequent experiments.

Moreover, we investigate the impact of sharing the student network $\Psi_{\text{stu}}^{\text{self}}$ to perform AFA and MPLT procedures during cyclic training. The experimental results, presented in Table XI, reveal that sharing $\Psi_{\text{stu}}^{\text{self}}$ for AFA and MPLT leads to a substantial parameter reduction of 40.2 M and achieves the optimal performance of 53.4% $mAP_{@50}$ compared with the decoupling configuration. This suggests that sharing a well-optimized student network for AFA and MPLT in a cyclic training manner effectively reinforces model performance while maintaining parameter efficiency.

F. Qualitative Results

1) Detection Result Visualization: We compare the results of different methods, including (a) Source-only [35], (b) MRT [21], (c) our proposed CMCT, and (d) Ground Truth, in Fig. 7. As illustrated, CMCT consistently delivers more accurate localization and classification. In the first row, CMCT successfully detects the entire train located at the center of the image, whereas the Source-only [35] and MRT [21] identify only fragments. This demonstrates the effectiveness of the CM strategy in capturing complete object-level semantic information from the target domain. Moreover, CMCT demonstrates its ability to uncover potential false negatives, as evidenced by the detection of a person behind the car in the second row and the car obscured by dense fog in the third row. Furthermore, CMCT effectively corrects misclassifications of foreground objects, as illustrated in the fourth row. Compared to MRT, which misclassifies a train as a bus and a car as a truck, our CMCT method produces results that closely align with the ground truth in the fourth column. These observations illustrate the effectiveness of CMCT.



(a) Source-only **(b) Source+Target** **(c) CMCT (Ours)**

Fig. 8: The t-SNE visualization of the source and target domains on Cityscapes→Foggy Cityscapes scenario. The blue and red points denote source and target object features, respectively.

2) T-SNE Visualization: We randomly sample an equal number of object queries from the source and target domains, and visualize them using the t-SNE method. Fig. 8 shows the visualization of these object queries in the Cityscapes→Foggy

Cityscapes adaptation scenario. Compared to the Source-only setting shown in Fig. 8 (a), introducing target domain data during pre-training alleviates the domain gap to some extent, as illustrated in Fig. 8 (b). Our proposed CMCT further mitigates the domain gap, as shown in Fig. 8 (c).

VI. Conclusion and Limitation

In this work, we present a novel Complementary Masking and Cyclic Training (CMCT) teacher-student framework for unsupervised domain-adaptive object detection. Specifically, we introduce a complementary masking strategy in the pre-training stage to help the teacher network acquire target domain knowledge, thereby enhancing the reliability of the generated pseudo-labels. In the self-training stage, we adopt a cyclic training strategy that alternates between mixed pseudo-label training and adversarial feature alignment. This strategy generates more reliable pseudo-labels while notably mitigating the negative impacts of low-quality ones. Experiments on three benchmark datasets validate the effectiveness of the proposed CMCT. Extensive ablation studies further demonstrate the contribution of each component in domain adaptation performance.

Although the proposed CMCT method achieves competitive performance against baseline approaches, it still faces several limitations. First, the training and inference processes are relatively time-consuming compared to lightweight CNN-based counterparts. Second, CMCT exhibits sensitivity to hyperparameters, where suboptimal parameter choices can hinder the learning of correct representations. Moreover, under severe domain shifts, such as from Daytime-Sunny to Night–Rain, the quality of pseudo-labels tends to degrade, thus deteriorating adaptation performance. In future work, we plan to investigate more robust vision–language models to mitigate these limitations and further enhance cross-domain generalization capability.

References

[1] C. Huang, Q. Xu, Y. Wang, Y. Wang, and Y. Zhang, "Self-supervised masking for unsupervised anomaly detection and localization," *IEEE Trans. Multimedia*, vol. 25, pp. 4426–4438, 2022.

[2] J. Wang, Z. Wu, D. Chen, C. Luo, X. Dai, L. Yuan, and Y.-G. Jiang, "Omnitracker: Unifying visual object tracking by tracking-with-detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2025.

[3] S. Vignesh, K. Karunya, V. S. KP, S. J. Shabu, and D. Poornima, "Obstacle detection on autonomous driving systems," in *ICoACT*. IEEE, 2025, pp. 01–04.

[4] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A theory of learning from different domains," *Mach. Learn.*, vol. 79, pp. 151–175, 2010.

[5] G. Wilson and D. J. Cook, "A survey of unsupervised deep domain adaptation," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 5, pp. 1–46, 2020.

[6] Y. Chen, W. Li, C. Sakaridis, D. Dai, and L. Van Gool, "Domain adaptive faster r-cnn for object detection in the wild," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun 2018. [Online]. Available: http://dx.doi.org/10.1109/cvpr.2018.00352

[7] Z. He and L. Zhang, "Multi-adversarial faster-rcnn for unrestricted object detection," in *Int. Conf. Comput. Vis.*, 2019, pp. 6668–6677.

[8] K. Saito, Y. Ushiku, T. Harada, and K. Saenko, "Strong-weak distribution alignment for adaptive object detection," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 6956–6965.

[9] D. Zhang, J. Li, L. Xiong, L. Lin, M. Ye, and S. Yang, "Cycle-consistent domain adaptive faster rcnn," *IEEE Access*, p. 123903–123911, Jan 2019. [Online]. Available: http://dx.doi.org/10.1109/access.2019.2938837

[10] H.-K. Hsu, C.-H. Yao, Y.-H. Tsai, W.-C. Hung, H.-Y. Tseng, M. Singh, and M.-H. Yang, "Progressive domain adaptation for object detection," in *IEEE Winter Conf. Appl. Comput. Vis.*, 2020, pp. 749–757.

[11] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," *Adv. Neural Inform. Process. Syst.*, vol. 30, 2017.

[12] J. Yu, J. Liu, X. Wei, H. Zhou, Y. Nakata, D. Gudovskiy, T. Okuno, J. Li, K. Keutzer, and S. Zhang, "Mttrans: Cross-domain object detection with mean teacher transformer," in *Eur. Conf. Comput. Vis.* Springer, 2022, pp. 629–645.

[13] Y. Li, X. Dai, C. Ma, Y. Liu, K. Chen, B. Wu, Z. He, K. Kitani, and P. Vajda, "Cross-domain adaptive teacher for object detection," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 7581–7590.

[14] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Int. Conf. Mach. Learn.* PMLR, 2015, pp. 1180–1189.

[15] M. Xu, H. Wang, B. Ni, Q. Tian, and W. Zhang, "Cross-domain detection via graph-induced prototype alignment," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 12 355–12 364.

[16] M. A. Munir, M. H. Khan, M. Sarfraz, and M. Ali, "Ssal: Synergizing between self-training and adversarial learning for domain adaptive object detection," *Adv. Neural Inform. Process. Syst.*, vol. 34, pp. 22 770–22 782, 2021.

[17] N. Inoue, R. Furuta, T. Yamasaki, and K. Aizawa, "Cross-domain weakly-supervised object detection through progressive domain adaptation," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5001–5009.

[18] Q. Cai, Y. Pan, C.-W. Ngo, X. Tian, L. Duan, and T. Yao, "Exploring object relation in mean teacher for cross-domain detection," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 11 457–11 466.

[19] J. Deng, W. Li, Y. Chen, and L. Duan, "Unbiased mean teacher for cross-domain object detection," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 4091–4101.

[20] M. Chen, W. Chen, S. Yang, J. Song, X. Wang, L. Zhang, Y. Yan, D. Qi, Y. Zhuang, D. Xie *et al.*, "Learning domain adaptive object detection with probabilistic teacher," *arXiv preprint arXiv:2206.06293*, 2022.

[21] Z. Zhao, S. Wei, Q. Chen, D. Li, Y. Yang, Y. Peng, and Y. Liu, "Masked retraining teacher-student framework for domain adaptive object detection," in *Int. Conf. Comput. Vis.*, 2023, pp. 19 039–19 049.

[22] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Int. Conf. Comput. Vis.*, 2017, pp. 2223–2232.

[23] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 16 000–16 009.

[24] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9729–9738.

[25] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Int. Conf. Mach. Learn.* PmLR, 2020, pp. 1597–1607.

[26] W. Weng and C. Yuan, "Mean teacher detr with masked feature alignment: a robust domain adaptive detection transformer framework," in *AAAI Conf. Artif. Intell.*, vol. 38, no. 6, 2024, pp. 5912–5920.

[27] G. Jin, F. Yang, M. Sun, R. Zhao, Y. Liu, W. Li, T. Bao, L. Wu, X. Zeng, and R. Zhao, "Seqco-detr: Sequence consistency training for self-supervised object detection with transformers," 2023.

[28] J. Hu, L. Qi, J. Zhang, and Y. Shi, "Domain generalization via inter-domain alignment and intra-domain expansion," *Pattern Recognit.*, vol. 146, p. 110029, 2024.

[29] J. Maurya, K. R. Ranipa, O. Yamaguchi, T. Shibata, and D. Kobayashi, "Domain adaptation using self-training with mixup for one-stage object detection," in *IEEE Winter Conf. Appl. Comput. Vis.* IEEE, 2023, pp. 4178–4187.

[30] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[31] Z. Chen, W. Zhang, X. Wang, K. Chen, and Z. Wang, "Mixed pseudo labels for semi-supervised object detection," *arXiv preprint arXiv:2312.07006*, 2023.

[32] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[33] S. Cao, D. Joshi, L.-Y. Gui, and Y.-X. Wang, "Contrastive mean teacher for domain adaptive object detectors," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 23 839–23 848.

[34] M. Kennerley, J. Wang, B. Veeravalli, and R. T. Tan, "Cat: Exploiting inter-class dynamics for domain adaptive object detection," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 16 541–16 550.
[35] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection. arxiv 2020," *arXiv preprint arXiv:2010.04159*, vol. 3, 2010.
[36] H. W. Kuhn, "The hungarian method for the assignment problem," *Nav. Res. Logist. Q*, vol. 2, no. 1-2, pp. 83–97, 1955.
[37] W. Chen, S. Li, J. Li, J. Yang, J. Paisley, and D. Zeng, "Dequantified diffusion-schrödinger bridge for density ratio estimation," *arXiv preprint arXiv:2505.05034*, 2025.
[38] W. Chen, S. Du, S. Li, D. Zeng, and J. Paisley, "Entropy-informed weighting channel normalizing flow for deep generative models," *Pattern Recognit.*, p. 112442, 2025.
[39] P. Yuan, W. Chen, S. Yang, Y. Xuan, D. Xie, Y. Zhuang, and S. Pu, "Simulation-and-mining: Towards accurate source-free unsupervised domain adaptive object detection," in *ICASSP*. IEEE, 2022, pp. 3843–3847.
[40] W. Wang, Y. Cao, J. Zhang, F. He, Z.-J. Zha, Y. Wen, and D. Tao, "Exploring sequence feature alignment for domain adaptive detection transformers," in *ACM Int. Conf. Multimedia*, 2021, pp. 1730–1738.
[41] W. Li, X. Liu, and Y. Yuan, "Scan++: Enhanced semantic conditioned adaptation for domain adaptive object detection," *IEEE Trans. Multimedia*, vol. 25, pp. 7051–7061, 2022.
[42] J. Zhang, J. Huang, Z. Luo, G. Zhang, X. Zhang, and S. Lu, "Da-detr: Domain adaptive detection transformer with information fusion," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 23 787–23 798.
[43] L. He, W. Wang, A. Chen, M. Sun, C.-H. Kuo, and S. Todorovic, "Bidirectional alignment for domain adaptive detection with transformers," in *Int. Conf. Comput. Vis.*, 2023, pp. 18 775–18 785.
[44] K. Wang, L. Pu, and W. Dong, "Cross-domain adaptive object detection based on refined knowledge transfer and mined guidance in autonomous vehicles," *IEEE T. Intell. Veh.*, vol. 9, no. 1, pp. 1899–1908, 2023.
[45] J. Deng, X. Zhang, W. Li, L. Duan, and D. Xu, "Cross-domain detection transformer based on spatial-aware and semantic-aware token alignment," *IEEE Trans. Multimedia*, vol. 26, pp. 5234–5245, 2023.
[46] Y. Liu, J. Wang, W. Wang, Y. Hu, Y. Wang, and Y. Xu, "Crada: Cross domain object detection with cyclic reconstruction and decoupling adaptation," *IEEE Trans. Multimedia*, vol. 26, pp. 6250–6261, 2024.
[47] J. Ke, L. He, B. Han, J. Li, D. Wang, and X. Gao, "Vldadaptor: Domain adaptive object detection with vision-language model distillation," *IEEE Trans. Multimedia*, 2024.
[48] H. Peng, D. Yang, M. Wang, W. Lin, and X. Zeng, "Guiding inter-domain class balancing with salient features for domain adaptive object detection," in *ICASSP*. IEEE, 2025, pp. 1–5.
[49] Z. Wen, J. Liu, H. Zhang, and F. Zuo, "Exploring fine-grained visual-text feature alignment with prompt tuning for domain-adaptive object detection," *IEEE T. Cybern.*, 2025.
[50] H. Li, R. Zhang, H. Yao, X. Zhang, Y. Hao, X. Song, and L. Li, "React: Remainder adaptive compensation for domain adaptive object detection," *IEEE Trans. Image Process.*, 2024.
[51] Y. Bai, Y. Wu, B. Zhu, and X. Li, "Contrastive-domain mean teacher for domain adaptive object detection," *IEEE Trans. Circuits Syst. Video Technol.*, 2025.
[52] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich, "Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks," in *Int. Conf. Mach. Learn.* PMLR, 2018, pp. 794–803.
[53] K. Deb, "Multi-objective optimisation using evolutionary algorithms: an introduction," in *Multi-objective evolutionary optimisation for product design and manufacturing*. Springer, 2011, pp. 3–34.
[54] O. Sener and V. Koltun, "Multi-task learning as multi-objective optimization," *Adv. Neural Inform. Process. Syst.*, vol. 31, 2018.
[55] P. Neal, C. Eric, P. Borja, and E. Jonathan, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.
[56] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3213–3223.
[57] C. Sakaridis, D. Dai, and L. Van Gool, "Semantic foggy scene understanding with synthetic data," *Int. J. Comput. Vis.*, vol. 126, pp. 973–992, 2018.
[58] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2636–2645.
[59] M. Johnson-Roberson, C. Barto, R. Mehta, S. N. Sridhar, K. Rosaen, and R. Vasudevan, "Driving in the matrix: Can virtual worlds replace human-generated annotations for real world tasks?" *arXiv preprint arXiv:1610.01983*, 2016.
[60] K. Gong, S. Li, S. Li, R. Zhang, C. H. Liu, and Q. Chen, "Improving transferability for domain adaptive detection transformers," in *ACM Int. Conf. Multimedia*, 2022, pp. 1543–1551.
[61] W. Li, X. Liu, and Y. Yuan, "Sigma: Semantic-complete graph matching for domain adaptive object detection," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 5291–5300.
[62] W. Li, X. Liu, and Y. Yuan, "Sigma++: Improved semantic-complete graph matching for domain adaptive object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 7, pp. 9022–9040, 2023.
[63] T. Huang, C. Huang, C. Ku, and J.-C. Chen, "Blenda: Domain adaptive object detection through diffusion-based blending," in *ICASSP*. IEEE, 2024, pp. 4075–4079.
[64] Z. Fu, C. Liu, Y. Chen, J. Zhou, Q. Liu, and Y. Wang, "De-simplifying pseudo labels to enhancing domain adaptive object detection," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–15, 2025.

Jing Teng (Senior Member, IEEE) received the B.Eng. degree in electronic information engineering from the Central South University, Changsha, China, in 2003 and the Ph.D. degree in systems optimization and reliability from the University of Technology of Troyes, Troyes, France, in 2009. Since 2010, she has been working in the School of Control and Computer Engineering, North China Electric Power University, Beijing, China. She is the author of more than 60 papers in refereed journals and international conference proceedings and has been serving as a reviewer of several international journals and conferences. Her research interests include artificial intelligence and its applications in energy system, transportation system etc.

Zhiwei Zhu obtained his B.S degree from Beijing Union University, Beijing, China, in 2019. He is currently pursuing the M.S. degree at the North China Electric Power University, with his primary research interests focusing on image segmentation and unsupervised domain adaptive object detection.

Xujie Long received the B.S. degree from North China Electric Power University, Beijing, China, in 2019, where he is currently pursuing the M.S. degree. His research focuses on image processing and semi-supervised object detection.

Mengyang Pu received the Ph.D. degree from the School of Computer and Information Technology, Beijing Jiaotong University, China, in 2022. In 2020, she was a visiting Ph.D. student in Computer Science at Stony Brook University. She is currently a lecturer with the School of Control and Computer Engineering, North China Electric Power University. Her research interests include computer vision, image processing, and deep learning.

Tian Wang (Senior Member, IEEE) received the bachelor's and master's degree from Xi'an Jiaotong University, Xi'an, China, in 2007 and 2010. He received the Ph.D. degree from the University of Technology of Troyes, Troyes, France, in 2014. He is currently a Professor with the School of Artificial Intelligence, Beihang University, Beijing, China. His research interests include pattern recognition and computer vision.

Hichem Snoussi received the Diploma degree in electrical engineering from École supérieure d'électricité, Gifsur-Yvette, France, in 2000, the DEA and Ph.D. degrees in signal processing from the University of Paris Sud, Orsay, France, in 2000 and 2003, respectively, and the HDR degree from the University of Technology of Compiégne, in 2009. From 2003 to 2004, he was a Post-Doctoral Researcher with the Institut de Recherche en Communications et Cybernétique de Nantes (IRCCyN). He has spent short periods as a Visiting Scientist with the Brain Science Institute, RIKEN, Japan, and the Olin Neuropsychiatry Research Center, Institute of Living, USA. From 2005 to 2009, he was an Associate Professor with the University of Technology of Troyes, France. Since 2010, he has been a Full Professor with the University of Technology of Troyes. Since 2010, he has been the in charge of CapSec Plateform (Sensors for Security). From 2016 to 2020, he has been the Manager of the LM2S Laboratory, Charles Delaunay Institute. Since 2021, he has been the Deputy Director of the LIST3N Laboratory, and the Team Leader of the M2S Group. He is the principal investigator of many research projects and industrial partnerships. His research interests include Bayesian techniques for source separation, information geometry, differential geometry, and machine learning.

Ruifeng Shi (Senior Member, IEEE) received the B.Eng., Msc.Eng. degree in aircraft design from Beijing University of Aeronautics and Astronautics, Beijing, China, in 1999 and 2002 respectively, and the Ph.D. degree in system engineering from Beihang University, Beijing, China, in 2006. He held a postdoctoral position with the School of Computer Science and Engineering from Beihang University, China, until 2008. He is now a full Professor with School of Control and Computer Engineering, North China Electric Power University, Beijing, China. His current research areas include V2G, integration/synergy of transportation and energy system, integrated energy system, energy trading and virtual power plant. Dr. Shi is a senior member of Chinese Computer Federation since 2011, a senior member of Chinese Society for Electrical Engineering since 2015, a member of committee experts on discrete system simulation, Chinese Association for System Simulation since 2011, a member of standing committee experts on integrated development of transportation and energy of China Highway Society since 2023, and the secretary-general of energy and transportation integration professional committee of China International Science and Technology Cooperation Association since 2025.

Y ou Lv (IEEE member) received the Ph.D. degree in North China Electric Power University, China. He is currently a professor of control science and engineering with North China Electric Power University, China. His main research interests include artificial intelligence and advanced energy system analysis, modeling and optimal control of energy storage systems, and intelligent power generation and flexibility regulation. He has coauthored over 50+ publications, including Applied Energy, Energy, Renewable Energy, Applied Thermal Engineering, ISA Transactions. He has supervised more than 10 master's/PhD students. He is the vice dean of the School of Control and Computer Engineering.

Jonathan Li (Fellow, IEEE) received the Ph.D. degree in geomatics engineering from the University of Cape Town, South Africa. He is currently a professor of geomatics and systems design engineering with the University of Waterloo, Canada. His main research interests include AI-based information extraction from earth observation images and LiDAR point clouds, pointgrammetry and remote sensing, GeoAI and 3D vision for digital twin cities and autonomous driving. He has coauthored over 650+ publications, including Remote Sensing of Environment, ISPRS Journal of Photogrammetry and Remote Sensing, International Journal of Applied Earth Observation and Geoinformation (JAG), IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING (TGRS), and in flagship conferences in computer vision and AI, including CVPR, NeurIPS, AAAI, and IJCAI. He has supervised more than 200 master's/PhD students as well as post-doctoral fellows/visiting scholars to completion. He is Editor-in-Chief of JAG and Associate Editor of TGRS. He is a Fellow of the Canadian Academy of Engineering, the Royal Society of Canada (Academy of Science), and the Engineering Institute of Canada. He is the President of the Canadian Institute of Geomatics (CIG).