Multi-View Domain Adaptive Object Detection on Camera Networks

Yan Lu[1], **Zhun Zhong**[2], **Yuanchao Shu**[3]

[1]New York University, [2]University of Trento, [3]Zhejiang University
jasonengineer@hotmail.com, zhunzhong007@gmail.com, ycshu@zju.edu.cn

Abstract

In this paper, we study a new domain adaptation setting on camera networks, namely Multi-View Domain Adaptive Object Detection (MVDA-OD), in which labeled source data is unavailable in the target adaptation process and target data is captured from multiple overlapping cameras. In such a challenging context, existing methods including adversarial training and self-training fall short due to multi-domain data shift and the lack of source data. To tackle this problem, we propose a novel training framework consisting of two stages. First, we pre-train the backbone using self-supervised learning, in which a multi-view association is developed to construct an effective pretext task. Second, we fine-tune the detection head using robust self-training, where a tracking-based single-view augmentation is introduced to achieve weak-hard consistency learning. By doing so, an object detection model can take advantage of informative samples generated by multi-view association and single-view augmentation to learn discriminative backbones as well as robust detection classifiers. Experiments on two real-world multi-camera datasets demonstrate significant advantages of our approach over the state-of-the-art domain adaptive object detection methods.

Introduction

Object detection aims at finding all regions of interests (RoIs) in an image and assigning each RoI to a semantic class. Recent works on object detection (Ren et al. 2015; Lin et al. 2017; Redmon and Farhadi 2018; Zhao et al. 2019) has achieved remarkable results on many public datasets. Nonetheless, the success is mainly attributed to supervised learning over large amounts of annotated data. Since the labor cost of RoI-level annotations is prohibitively expensive, domain adaptive object detection (DA-OD) algorithms (Zhuang et al. 2020; He and Zhang 2019) have been developed in various scenarios (*e.g.*, adverse weather conditions (Sakaridis, Dai, and Gool 2018; Nada et al. 2018; Li et al. 2019), synthetic data adaptation (Matthew et al. 2017; Inoue et al. 2018), and cross-camera adaptation (Cordts et al. 2016; Yu et al. 2020; Geiger et al. 2013)) to adapt models from labeled data (*a.k.a.*, the source domain) to unlabeled data (*a.k.a.*, the target domain).

Despite the advancements in DA-OD, most existing methods require access to the source domain during the adaptation phase. However, due to privacy and compliance issues, companies and organizations that have large-scale labeled data are commonly reluctant to share their data with users who want to adapt the model to their own environments. Instead, they provide users with models pre-trained on the labeled data. Moreover, existing methods commonly focus on the setting of a single target domain and lack design on simultaneous adaptation to multiple target domains. On camera networks, however, videos are usually captured by multiple cameras (views) with overlapping fields of view, which can be regarded as different but non-independent domains. Existing methods will produce inferior performance in this context due to neglecting spatial-temporal correlations and cross-camera domain shifts.

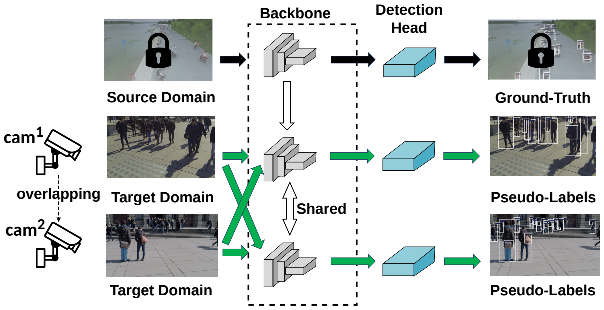

Figure 1: An overview of MVDA-OD. All cameras train a shared backbone with multi-view fusion, and each camera fine-tunes its detection head with its local data and pseudo-labels.

To this end, this paper focuses on a more practical setting on camera networks where Multi-View Domain Adaptive Object Detection (MVDA-OD) is desired. In MVDA-OD, a fleet of cameras with overlapping views share their unlabeled data and train their backbones of object detection models collaboratively. Conceptually, backbones need to generalize on unseen domains while detection heads are better positioned to learn domain-specific features on each target domain. Thus, we propose a two-stage adaptation framework for MVDA-OD, as depicted in Figure 1. Unlike DA-OD, MVDA-OD only requires the pre-trained model and is designed to learn camera-specific object detection models from new data captured by cameras with overlapping fields of view.

Intuitively, extending existing DA-OD to MVDA-OD can be achieved by 1) adversarial training of a single model on combined data from multiple target domains (Roy et al. 2021; Isobe et al. 2021), or 2) training multiple models for each target domain using pseudo-labels and a self-paced learning paradigm (Jiang et al. 2015). However, they both fall short in practice. Although adversarial-based approaches are effective to learn domain-invariant features of all domains, its assumption that source data is available during adaptation does not hold in MVDA-OD. Self-paced based methods (RoyChowdhury et al. 2019; Kim et al. 2019; Khodabandeh et al. 2019; Li et al. 2021a), which are also called self-training widely used in semi-supervised learning (Gao et al. 2019; Jeong et al. 2019; Verma et al. 2019; Yang et al. 2021b), create pseudo-labels for unlabeled images using a pre-trained model, and jointly train a model with both labeled and unlabeled data (RoyChowdhury et al. 2019; Kim et al. 2019; Khodabandeh et al. 2019; Li et al. 2021a). In a real deployment, however, they suffer from overfitting (Yang et al. 2021b) as there is only **little unlabeled data** on a single camera. However, most self-training approaches (RoyChowdhury et al. 2019; Kim et al. 2019; Khodabandeh et al. 2019) require a large amount of unlabeled data to refine pseudo-labels. Without removing incorrect pseudo-labels, end-to-end self-training methods are easy to overfit on noisy data. Furthermore, pseudo-labels generated from a single pre-trained detection head are often noisy and could easily lead to training collapse.

To resolve the overfitting issue, in MVDA-OD, we design a novel and effective self-supervised learning approach to pre-train the backbone through multi-view association. In contrast to end-to-end self-training methods (RoyChowdhury et al. 2019; Kim et al. 2019; Khodabandeh et al. 2019), we start by pre-training a robust feature extractor for downstream head fine-tuning because backbone layers are hard to learn discriminative features with a small number of noisy pseudo-labels. Specifically, we construct a multi-view reID pretext task which makes it easier for the feature extractor to learn representative features of new scenarios because of various viewpoints provided by multi-view fusion.With a trained backbone, a fine-grained detection head of each view is learned in the second stage. Motivated by recent works on self-training with consistency (Berthelot et al. 2020; Yang et al. 2020) that utilize augmentation and consistency regularization to enhance the stability of the self-training process, we propose a robust single-view self-training approach. It leverages off-the-shelf tracking techniques to augment single-view viewpoints for a predicted bounding box and fine-tunes detection heads via weak-hard consistency learning.

In summary, this paper makes four contributions: **1)** a new and practical setting for domain adaptive object detection, *i.e.*, MVDA-OD; **2)** a novel two-stage training framework for MVDA-OD; **3)** two effective training approaches in this framework; and **4)** the state-of-the-art performance on WildTrack (72.50% and 72.41%) and CityFlow (69.49% and 69.82%) on pre-trained YOLOv3 and Faster R-CNN.

Related Work

Domain adaptive object detection seeks to adapt a robust object detector from labeled source data to unlabeled target data. Most existing works (Chen et al. 2018b; Zheng et al. 2020a; Munir et al. 2021; Wang et al. 2021) adopt adversarial feature learning (Ganin and Lempitsky 2015) and build two domain alignments or classifiers to let backbone and detection head extract image-level and instance-level (or bounding box level) domain-invariant features. To align accurately, an effective two-stage framework (Munir et al. 2021) is proposed, which leverages uncertain pseudo-labels to train backbone layers in a self-supervised way and then uses it to find more accurate areas for alignment. Benefiting from refinement, adversarial feature learning can be easy to learn domain-invariant features. They need data from source and target domains to train models. In real-world scenarios, gaining access to source data might not be feasible due to privacy concerns, legal issues, and limited network bandwidth. To disengage from source data, recent studies (RoyChowdhury et al. 2019; Khodabandeh et al. 2019; Kim et al. 2019; Li et al. 2021b; Yang et al. 2021a) follow a self-training framework and train models on target data independently. Specifically, they generate pseudo-labels for a target domain and train models in a supervised learning way. Although self-training methods outperform adversarial-based approaches in many datasets, they must refine pseudo-labels with numerous unlabeled target domain samples. Unfortunately, most target domains in real-world applications do not have enough unlabeled data to support iterative refinement. To improve generalization ability with practical constraints, we devise a novel two-stage training framework to adapt pre-trained object detection models to multiple target domains, where the shared backbone learns discriminative features through multi-view self-supervised learning and detection heads learn robust classifiers on each domain by weak-hard consistency learning.

Multi-target domain adaptation (MTDA) aims to transfer knowledge from a single labeled dataset to multiple unlabeled target datasets. Unlike single-target domain adaptation, MTDA needs to extract effective domain-invariant features for all domains. The mismatching issue across target domains degrades the performance of the classic adversarial-based adaptation framework (Ganin and Lempitsky 2015). To address this problem, some recent works (Roy et al. 2021; Isobe et al. 2021) develop a novel aggregation strategy. In classification, for instance, D-CGCT (Roy et al. 2021) utilizes a graph convolutional network to aggregate features across different domains and develops a co-teaching strategy to avoid the overfitting issue. In segmentation, CCL (Isobe et al. 2021) adds the collaborative consistency regularization term to the adversarial adaptation phase on each target domain. However, both require labeled source data during adaptation and ignore correlations between target domains. Our work studies a new setting, namely multi-view domain adaptive object detection, where labeled source data is unavailable during the adaptation phase, and the target data is captured by multiple overlapping cameras[1].

[1] Please refer to Appendix 1 for multi-view object detection.

Methodology

Problem Definition. In the multi-view domain adaptation setting, we are given an object detection model pre-trained on labeled source data S and unlabeled target data T from M target domains $T^1, T^2, ..., T^M$. Each target domain $T^i = \{X_i^j\}_{j=1}^{N_i}$ is captured from an individual surveillance camera C^i and there exists overlapping field of view between cameras. X_i^j represents the j^{th} unlabeled frame and N_i is the number of unlabeled images in T^i. The goal of MVDA-OD is to adapt the pre-trained source model to the multi-view target domains. In this setting, two unique factors differ from traditional unsupervised domain adaptive object detection. First, the labeled source data is not available during the adaptation process, and only the pre-trained source model is provided. Second, the target data is formed by multiple target domains with strong spatial-temporal correlations. In the following, we take two overlapping cameras (*i.e.*, M=2) as an example to introduce the proposed method.

Overview of the Framework

In Figure 2, we show the framework of our method, which includes two training stages: (1) multi-view feature extractor learning and (2) single-view detection head learning. The first step aims to learn discriminative representation for objects in the target domains using self-supervised learning with multi-view association. The second step aims to learn an accurate detection classifier by robust self-training with single-view augmentation and weak-hard consistency learning.

Self-Supervised Learning with Multi-View Association

Motivation. The feature extractor F in object detection is responsible for extracting discriminative features. However, existing training methods (adversarial training and self-training) are limited by domain shift and limited data in MVDA-OD. It is because large-scale unlabeled video data is hard to collect with a single camera. Thus, an intuitive idea is leveraging all video data from camera networks to pre-train F in a self-supervised learning manner. Moreover, overlapping views of camera networks provide many effective tools (*e.g.,* epipolar geometry) to find an effective pretext task for pre-training.

Multi-View reID Data Generation. In the proposed multi-view self-supervised learning approach, one important step is to generate pairs of images with bounding boxes, where we hope each pair of images belongs to the same identity. To achieve this goal, we propose to associate bounding boxes by re-identification (reID) technique (Zheng et al. 2020b; He et al. 2020). Since there are no annotations, we can not learn reID models on the target data in a supervised manner. Although we can borrow existing pre-trained public reID models, we still meet two challenges in MVDA-OD. First, existing reID models (Zheng et al. 2020b; He et al. 2020) trained on their own datasets commonly produce low reID performance on the data of our setting, due to the large domain gap between datasets. As a result, we will generate a decent number of false positive pairs and thus seriously damage the following self-supervised learning process. Second, pairwise comparison between bounding boxes among all cameras incurs a non-linear computation overhead, which is prohibitively high for scenarios with busy traffic. To deal with these two challenges, we present a prune-and-augment approach, which first filters out a large number of bounding boxes that are less likely to be confirmed by reID using epipolar constraints (Zhang 1998), and then augments refined associated pairs through tracking.

In multi-view association, we begin from running an off-the-shelf or pre-trained object detection model on all frames for each camera. Outputs from detection contain bounding boxes which represent the location of RoIs (bounding boxes) and the corresponding predicted classification score (c). Like other works in video analytics, we filter bounding boxes whose c are smaller than c_{thr}. Based on the detection results, we extract an bounding box (x_1^i) from the i^{th} frame on T^1 to show multi-view association. To associate x_1^i with bounding boxes in X_2^i, we first use epipolar constraints of stereo vision to draw an epipolar area for x_1^i on X_2^i [2]. Then, we extract predicted bounding boxes in the epipolar area on X_2^i as candidate bounding boxes for x_1^i. Finally, we use a pre-trained reID model to extract features for all candidate bounding boxes and sort them by cosine distance in ascending order. Because each bounding box only has one associated bounding box in the other view, we select bounding box with the minimal distance as the associated bounding box for x_1^i. As Illustrated in Figure 3, we are able to significantly reduce the search space because the epipolar area helps us filter many bounding boxes before running a reID model on all candidate bounding boxes in X_2^i. In our implementation, we use SBS (He et al. 2020) and VehicleNet (Zheng et al. 2020b) for person and vehicle reID, respectively. Note that extending two views to multi-views is straightforward. One can simply follow the same searching pipeline to find candidate bounding boxes for x_1^i in all views. With multi-view searching, we increase the training data size from different viewpoints.

Nonetheless, it only finds out pairs of bounding boxes on cameras at the same time (associating bounding box for x_1^i on X_2^i). To augment associated bounding boxes in consecutive frames, we run a tracking model on them to add new bounding boxes of the same identity. Then we filter repeated identities that have the same bounding boxes. In an implementation, we run SiamMask-E (Xin and K 2019), a tracking algorithm on subsequent ten frames from both cameras to get more associated bounding boxes.

Consistency Training. With the multi-view reID data, F takes pairs of images with bounding boxes belonging to the same object as input and runs them through the entire model to get predicted distributions. Afterward, it calculates the classification score (cls) of each image purely based on features within the paired bounding box, and uses consistency loss in backpropagation to train the backbone network (Figure 2). Given a mini-batch of images from T^1 and T^2, consistency loss is defined as:

$$L^F_{consistency} = \sum_{k=1}^{N_{group}} CE(F(x_1^k), F(x_2^k)), \qquad (1)$$

where N_{group} is the total number of group of associated

[2] Details about epipolar geometry can be found in Appendix 7.

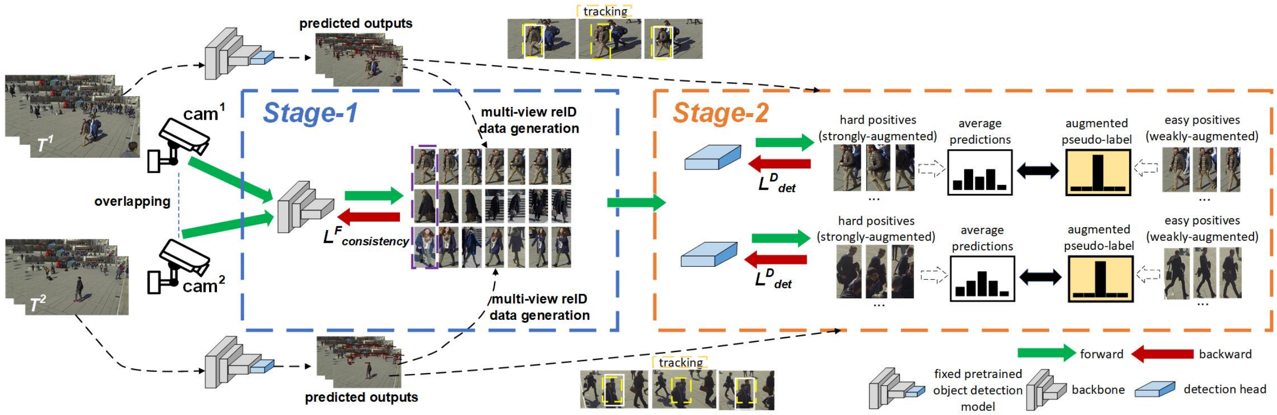

Figure 2: An overview of our two-stage training strategy. In the first stage, we build multi-view reID data with pre-trained detection and reID models, which are then used to fine-tune the backbone module (F) in a self-supervised manner. In the second stage, we adopt tracking techniques to mine easy and hard positives (missing bounding boxes) and train a detection head (D) in a robust self-training approach through weak-hard consistency learning.

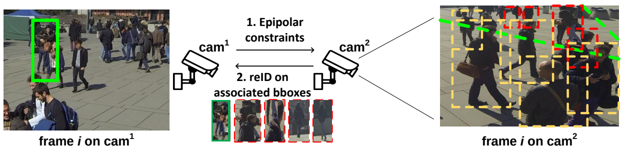

Figure 3: Illustration of epipolar mapping and reID. In cam^1, the green solid rectangle denotes an given bounding box (x_1^i). In cam^2, yellow and red dotted rectangles denote predicted bounding boxes. Two green dotted lines represent epipolar constraints on x_1^i.

bounding boxes in the mini-batch, CE represents cross entropy function, $F(\cdot)$ represents the classification model (the feature extractor F and a classification head), x_1^k and x_2^k denote the k^{th} pair of associated bounding boxes on T^1 and T^2 respectively. As we compute consistency loss between any pair of bounding boxes in Equation. 1, any RoIs detected by two or more views are selected as training data. As consistency loss is minimized by fine-tuning, the feature extractor F generates more representative feature maps. In an implementation, we add one fully connected (FC) layer to F and use it to generate predicted class distributions. After training, we delete this FC layer and save the feature extractor F for stage 2.

Discussion. A prevailing training strategy for self-learning methods (Khodabandeh et al. 2019; Li et al. 2021a,b; Yang et al. 2021a) is end-to-end iterative training with refined pseudo-labels. Although they are effective on many unlabeled datasets, pseudo-label mining depends on data size largely. In our setting, a single camera often cannot save many images for pseudo-label mining due to a memory constraint. Thus, a simple but effective approach is to split the fine-tuning process into two stages, which makes noisy pseudo-labels unable to distort pre-trained features and avoids overfitting issues on limited data for backbone layers simultaneously. We verify the effectiveness of our method in Table 2.

Robust Self-Training with Single-View Augmentation

Motivation. Self-training methods are widely used to adapt a detection head for each camera. But in MVDA-OD, limited data makes them hard to mine clean pseudo-labels. Thus, we leverage a new effective training approach named FixMatch (Berthelot et al. 2019; Yang et al. 2020), which improves the robustness of self-training via entropy minimization. Specifically, it uses a weakly-augmented example to generate an artificial label for a sample and enforce consistency against its strongly-augmented counterpart. But it is hard to find useful weakly-/strongly-augmentation for MVDA-OD. Fortunately, we observe that a tracking detector can find "hard" samples for an object, which commonly has a large difference in pose and viewpoint to its query counterpart. Thus, we propose a viewpoint-aware augmentation approach to adapt a robust head for each camera.

Augmented Pseudo-labels Construction. Before self-training, we first use a pre-trained tracking model to generate movement's trace for a given bounding box. Second, we select bounding boxes that are not detected in the current frame and are detected in consecutive frames as hard positives. It's because they are "missing" bounding boxes for object detection models. After splitting hard positives, we group the remaining bounding boxes which are detected by both detection and tracking models, as easy positives. Figure 4 illustrates running object detection and tracking algorithms on three consecutive frames. As there are existing matching detections in adjacent frames for the current tracking object which is not detected, it is correctly considered to be a "missing detection" and we add them to the set of hard positives. In contrast, samples that are found by detection and tracking algorithms concurrently are added to the set of easy positives. To reduce negative impacts caused by occlusion, we only keep positives whose distances between themselves and the selected bounding box on feature maps are smaller than T_{thr}.

Self-Training with Augmented Pseudo-Labels. With different groups of positives, we first build the augmented

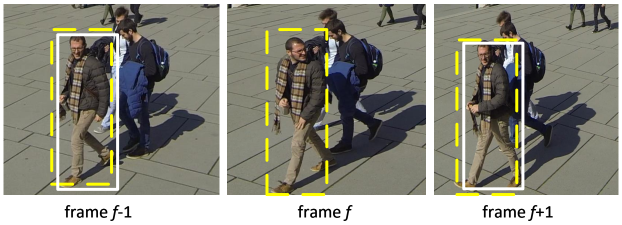

Figure 4: Illustration of Positives Mining. The solid white boxes denote detections, and the dashed yellow boxes are associated with the tracking algorithm.

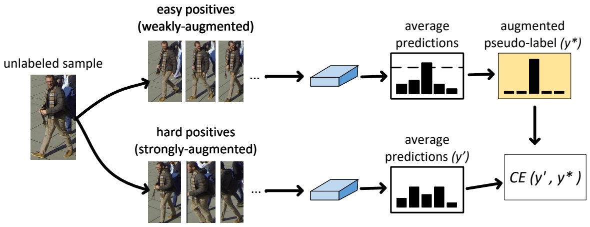

Figure 5: Diagram of self-training with augmented pseudo-labels. $CE(\cdot)$ represents the cross entropy functions. 'average predictions' are calculated by averaging all predictions on easy/hard positives. y and y^* denote the average prediction on hard positives and the augmented pseudo-label on easy positives, respectively. The classification categories depend on the pre-defined classes.

pseudo-label for a given input bounding box through average ensemble predictions on all easy positives. Second, we train D by making its prediction on all hard positives match the augmented pseudo-label via a cross-entropy loss. As shown in Figure 5, given an predicted bounding box (x), the loss L_{det}^{D} can be formulated as:

$$L_{det}^{D} = CE(\frac{1}{N_h}\sum_{i=1}^{N_h} D(x_h^i), O(\frac{1}{N_e}\sum_{j=1}^{N_e} D(x_e^j))), \qquad (2)$$

where N_h and N_e denote the number of hard and easy positives, respectively. x_h^i and x_e^j represent the i_{th} hard and j^{th} easy positives, respectively. $O(\cdot)$ denotes $\arg\max(\cdot)$.

In summary, our proposed training framework contains two stages: 1) backbone pretraining with mult-view data that aims to learn a shared discriminative image-level feature extractor for all cameras; 2) detection head fine-tuning with augmented pseudo-labels that focuses on learning a robust bounding box classifier for each camera.

Experiments

Experimental Setup

Datasets. In this paper, we conduct experiments on one general dataset (MS-COCO (Lin et al. 2014)) and two real-world multi-camera datasets (WildTrack (Chavdarova et al. 2018) and CityFlow (Tang et al. 2019)[3]) for evaluating the proposed method under the introduced MVDA-OD setting. MS-COCO is a large-scale object detection dataset that includes 330K images of 80 object categories. We regard it as the source domain for pre-training the source model. WildTrack is by far the largest multi-camera dataset for pedestrian detection and tracking. CityFlow is built for multi-camera vehicle tracking. These two datasets are used as the target domain. The details of them are shown in Table 1.

Implementation Details. We select YOLOv3 (Redmon and Farhadi 2018) with a backbone of Darknet53 and Faster R-CNN (Ren et al. 2015)[4] with a backbone of ResNet101 as the object detection models, which are implemented with mmdetection (Chen et al. 2019) toolbox. The detection models are first pre-trained on MS-COCO and then adapted to WildTrack and CityFlow with the proposed method, respectively. For evaluation, we set the ratio of the training set, evaluation set, and testing set to $16:4:5$ for both WildTrack and CityFlow. We adopt SiamMask-E (Xin and K 2019) as the tracking model in our method. SBS (He et al. 2020) and VehicleNet (Zheng et al. 2020b) are used for person and vehicle reID, respectively. In pseudo-label construction, we set T_{thr} to 0.7. During training, we choose Adam (Kingma and Ba 2015) as the optimizer and set the learning rate to 0.01. The batch size is set to 8. The threshold of classification score (c_{thr}) is set to 0.5. We train the model with 60 epochs in total, in which self-supervised multi-view training and single-view detection head fine-tuning are trained with 30 epochs individually.

Competitors. We compare the proposed method with the source-only model, three self-training approaches, and two domain adaptive object detection approaches: Source-Only, Self-Training (ST) (Gao et al. 2019), Self-Training with Gold Loss Correction (ST-GLC) (Dan et al. 2018), Self-Training with Consistency Loss (ST-CL) (Jeong et al. 2019), Self-Training with Hard Samples (ST-HARD) (RoyChowdhury et al. 2019), Domain adaptive Faster RCNN (DA-FR) (Chen et al. 2018a), Image-Instance Full Alignment Networks (DA-iFAN) (Zhuang et al. 2020), Vector-Decomposed Disentanglement (DA-VDD) (Wu et al. 2021), Similarity-based Domain Alignment (DA-SDA) (Rezaeianaran et al. 2021) and RPN Prototype Alignment (DA-RPN) (Zhang, Wang, and Mao 2021)[5] (see Appendix 2 for details).

Evaluation Metric. We use mean Average Precision (mAP) over Intersection Over Union (IoU) of 0.5 (mAP@[0.5:1.0]) to measure the detection performance on the target dataset. Due to space limitations, we only report the average mAP of all cameras in this section. Results on each camera can be found in Appendix 4.

Analysis on Two-Stage Training Framework

To validate the effectiveness of the proposed two-stage training framework, we compare four training strategies: (1) the proposed two-stage training strategy, which first pre-trains the backbone by self-supervised learning and then fine-tunes the detection head by robust self-training; (2) variation of our two-stage training, which trains the whole model (back-

[3] We used data collected from the first intersection of CityFlow.

[4] Due to page restrictions, we move all experimental results of Faster R-CNN to Appendix 3.

[5] Because DA-RPN requires to align features on RPN layers and one-stage object detection models don't have RPN, we only report the corresponding results for Faster R-CNN.

	Objects	# Cam	Size	Frames (labeled/total)	Avg. obj./frame	Category
WildTrack	Pedestrians	7	1920*1080	400 / 29400	23	Person
CityFlow	Vehicles	5	960*480	640 / 9775	13	Car, Truck and Bus

Table 1: Dataset statistics for WildTrack and CityFlow.

Backbone Pre-training	Self Training	WildTrack	CityFlow
×	$F+D$	65.23	61.56
×	D	63.15	62.27
✓	$F+D$	66.54	63.92
✓	D	**72.50**	**69.49**

Table 2: Average mAP (%) of 4 training strategies with YOLOv3 for 7 cameras in WildTrack and 5 cameras in CityFlow. In ”Backbone pre-training” column, ”✓” and ”×” represent the backbone pre-training is whether or not used in the training strategy. In ”Self-Training” column, ”$F+D$” denotes that fine-tuning backbone and detection heads together in stage 2. ”D” represents that we only fine-tune detection heads in stage 2.

Backbone Pre-training	Self Training	Epipolar Constraints	Extra reID cost	WildTrack	CityFlow
Self-training	$F+D$	×	0.00	62.45	61.13
Self-training	D	×	0.00	58.23	55.12
Self-supervised	D	×	100.00/100.00	69.84	71.22
Self-supervised	D	✓	**12.34/10.12**	**72.50**	**69.49**

Table 3: Average mAP (%) of self-training and self-supervised learning strategies with YOLOv3 for 7 cameras in WildTrack and 5 cameras in CityFlow. In ”Epipolar Constraints” column, ”✓” and ”×” represent the epipolar constraints is whether or not used in backbone pre-training. In ”Extra reID cost (%)” column, we record the frequency of running a reID model in stage 1 and then compute its corresponding probability. Without epipolar constraints, we have to run a reID model on all paired bounding boxes. Thus, we set it to the maximum number of running a reID model.

bone and detection head) by robust self-training in stage-2; (3) one-stage training, which trains the detection head with robust self-training; (4) variation of one-stage training, which trains the whole model with robust self-training.

In Table 2, we compare the results of these four strategies for WildTrack and CityFlow. The mAP averaged on all cameras is reported. We have the following conclusions. *First*, no matter using which self-training strategies (training the whole model or only the detection head), the two-stage training strategy can consistently produce higher performance than the one-stage strategy. This demonstrates the importance of the proposed self-supervised multi-view training, which helps the model to learn more discriminative representation for the downstream detection task. *Second*, when implementing self-training, optimizing the whole model commonly produces lower results than optimizing the detection head only (except for one case in YOLOV3 of WildTrack). This indicates that using a limited number of samples, which are assigned pseudo-labels, cannot support the backbone to learn representative features. The main reason is that training with few samples may lead to the model overfitting on them and thus decreases the discrimination of the backbone. *Third*, the proposed two-stage training strategy largely outperforms the other three strategies, validating the effectiveness of learning powerful backbone by self-supervised learning and learning robust detection head by self-training.

Evaluation on Multi-View Self-Supervised Learning

Self-Training vs. Self-Supervised Learning. To show the effectiveness of our self-supervised learning in stage 1, we compare it with a naive self-training strategy that trains object detection models with pseudo-labels in an end-to-end manner. For a fair comparison, the self-training pipeline also consists of two stages: we first train detection models on multi-view unlabeled data and then fine-tune them on a single view. As illustrated in Table 3, self-training achieves a much lower accuracy than our method. This is mainly due to two reasons. First, despite the increased amount of data, training samples from different cameras are still too noisy, hampering the model to learn a good feature representation and a detection head. In specific, the accuracy of fine-tuning detection heads in stage 2 only is much lower than that of fine-tuning the whole detection model (58.23% mAP vs. 62.45% mAP). To verify the effectiveness of epipolar constraints in stage 1, we record extra reID cost and compare it with running a reID model on all paired bounding boxes. Results are also shown in Table 3. Interestingly, epipolar constraints not only reduce the extra reID cost largely but also improve final mAP moderately. It is because pre-trained reID models have domain shifts on target datasets, and they are hard to provide correct predictions on all paired bounding boxes.

Comparison of Different Pretext Tasks. In this work, we propose to use multi-view reID as the pretext task for self-supervised backbone learning. To demonstrates its advantage, we further compare it with three popular pretext tasks, *i.e.*, rotation (Gidaris, Singh, and Komodakis 2018) (predicting which rotation has been applied for an input image), colourisation (Zhang, Isola, and Efros 2016) (predicting which the mapping quantized color value has applied for an input image) and relative position (Doersch, Gupta, and Efros 2015) (predicting the relative position between two random patches from one image). Results are reported in Table 4. Without using the pretext task, we directly fine-tune the detection head with our robust self-training. We can observe that the compared three pretext tasks fail to improve the performance in most settings. This indicates that these three pretext tasks can not help the model to learn more discriminative representation for MVDA-OD. We also compare SimCLR (Chen et al. 2020), a recent popular contrastive learning framework, with our method. Interestingly, SimCLR is hard to learn effective single-class detection models for WildTrack and obtains worse performance than relative position in multi-class detection for CityFlow. This is because SimCLR aims to learn inter-class inconsistency but ignores intra-class inconsistency which is important to classify on a small number of similar categories. In contrast, our proposed multi-view reID significantly improves the performance on all settings, verifying the large advantage of our designed pretext task for MVDA-OD.

Pretext Task	WildTrack	CityFlow
N/A	63.15	62.27
Rotation	60.11	55.45
Colourisation	63.15	61.44
Relative position	62.34	59.27
SimCLR	60.58	61.23
Multi-view reID	**72.50**	**69.49**

Table 4: Comparison of different pretext tasks with YOLOv3 in self-supervised backbone pre-training. mAP averaged on all cameras is reported.

Evaluation on Robust Self-Training

Comparison of Different Augmentation Methods. To evaluate the effectiveness of our proposed tracking-based augmentation method, we compare it with four widely used augmentation methods, including Brightness, color, contrast and autoaugment (Cubuk et al. 2019). Following FixMatch (Yang et al. 2020), we use filp-and-shift as weak augmentation and the four approaches as strong augmentation. In addition, we also conduct experiments on vanilla self-training which does not use any augmentation during training. Results in Table 5 show that the compared four augmentations can slightly improve the performance in some settings but also will reduce the performance in other settings. This indicates that these four augmentation methods can not achieve consistent improvement in all settings. Instead, our proposed tacking-based augmentation method leads to large improvements in all settings, which significantly outperforms the compared three augmentation methods. This shows the importance of considering the view variations for learning robust detection head and verifies the advantage of our tracking-based augmentation in MVDA-OD.

Strong Augmentation	WildTrack	CityFlow
×	65.41	62.15
Brightness	62.48	61.58
Color	61.15	60.38
Contrast	66.28	62.08
AutoAugment	63.79	64.56
Tracking (ours)	72.50	69.49

Table 5: Comparison of different augmentation methods with YOLOv3 in robust self-training. mAP averaged on all cameras is reported.

Comparison with State-of-the-Art Methods

In Table 6, we compare the proposed method with 9 state-of-the-art (SOTA) methods on WildTrack and CityFlow, including 4 self-training approaches (ST, ST-GLC, ST-CL and ST-HARD) and 5 domain adaptation methods (DA-FR, DA-iFAN, DA-VDD and DA-SDA). For self-training approaches, we keep the source-free constraint of MVDA-OD. That is, we only use the source pre-trained model and target domain to implement self-training approaches. For domain adaptation methods, we remove the source-free constraint and apply them by jointly training the model with both source data and target data. We have the following observations. First, existing SOTA methods fail to achieve clear improvement, or even will reduce the performance, on both datasets. Importantly, even using the source data during adaptation process, the DA-FR, DA-SDA and DA-iFAN still produce poor results compared to the source-only model. This shows that existing self-training methods and domain adaptation methods are limited by the source-free and multi-camera constraints and thus are not suitable for the proposed MVDA-OD. Second, the average mAP of our method on all cameras is significantly higher than all methods, regardless of the data set and detection model. Specifically, when testing on WildTrack, our approach outperforms the best known self-training approach (ST-HARD) by 5.34% mAP for YOLOv3. Also, our method is higher than the best domain adaptive method (DA-SDA) by 4.31% mAP for YOLOv3. A similar superiority of our method can also be found when testing on CityFlow. Third, our method produces slightly lower results on CAM-7 when testing on WildTrack. The main reason is that CAM-7 has very limited shared field of view (FOV) with other cameras and thus very few non-camera-specific training data can be discovered for multi-view self-supervised learning. Comparisons on each camera and results on SOTA methods using all multi-view data can be found in Appendix 5 and Appendix 6, respectively.

Method	Source Data	WildTrack	CityFlow
Source-Only		64.11	61.36
ST		63.43	55.30
ST-CL		63.68	55.85
ST-GLC		65.95	57.22
ST-HARD		67.16	57.03
DA-FR	✓	63.16	55.26
DA-iFAN	✓	64.66	56.53
DA-VDD	✓	65.32	57.21
DA-SDA	✓	68.19	59.13
Ours		**72.50**	**69.49**

Table 6: Comparison with SOTA methods on WildTrack and CityFlow using YOLOv3. In ”Source Data” column, ”✓” denotes that the source data is available during the training phase.

Conclusion

In this paper, we study a new and more practical setting (MVDA-OD) for existing domain adaptive object detection. Unlike other source-free settings, we point two unique challenges for MVDA-OD: 1) the limited variety spaces of target data is too small to train effective feature extractors; and 2) coarse and noisy pseudo-labels are easy to lead to overfitting issues; To address these, we propose a novel two-stage training framework: 1) learning shared discriminative feature extractor in a novel self-supervised manner with multi-view reID pretext task and 2) learning robust detection classifiers through weak-hard consistency learning. Extensive experiments are conducted on two real-world multi-camera datasets and our method obtains the state-of-the-art results.

References

Berthelot, D.; Carlini, N.; Cubuk, E. D.; Kurakin, A.; Sohn, K.; Zhang, H.; and Raffel, C. 2020. ReMixMatch: Semi-Supervised Learning with Distribution Alignment and Augmentation Anchoring. In ICLR.

Berthelot, D.; Carlini, N.; Cubuk, E. D.; Kurakin, A.; Zhang, H.; and Raffel, C. 2019. Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. In ICLR.

Chavdarova, T.; Baque, P.; Bouquet, S.; Maksai, A.; Jose, C.; Lettry, L.; Fua, P.; Gool, L. V.; and Fleuret, F. 2018. The WILDTRACK Multi-Camera Person Dataset. In CVPR.

Chen, K.; Wang, J.; Pang, J.; Cao, Y.; Xiong, Y.; Li, X.; Sun, S.; Feng, W.; Liu, Z.; Xu, J.; Zhang, Z.; Cheng, D.; Zhu, C.; Cheng, T.; Zhao, Q.; Li, B.; Lu, X.; Zhu, R.; Wu, Y.; Dai, J.; Wang, J.; Shi, J.; Ouyang, W.; Loy, C. C.; and Lin, D. 2019. MMDetection: Open MMLab Detection Toolbox and Benchmark. arXiv:1906.07155.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A Simple Framework for Contrastive Learning of Visual Representations. In ICML.

Chen, Y.; Li, W.; Sakaridis, C.; Dai, D.; and Gool, L. V. 2018a. Domain Adaptive Faster R-CNN for Object Detection in the Wild. In ICCV.

Chen, Y.; Li, W.; Sakaridis, C.; Dai, D.; and Gool1, L. V. 2018b. Domain Adaptive Faster R-CNN for Object Detection in the Wild. In CVPR.

Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; and Schiele, B. 2016. The Cityscapes Dataset for Semantic Urban Scene Understanding. In CVPR.

Cubuk, E. D.; Zoph, B.; Mane, D.; Vasudevan, V.; and Le, Q. V. 2019. AutoAugment: Learning Augmentation Strategies from Data. In CVPR.

Dan, H.; Mantas, M.; Duncan, W.; and Kevin, G. 2018. Using Trusted Data to Train Deep Networks on Labels Corrupted by Severe Noise. In NeurIPS.

Doersch, C.; Gupta, A.; and Efros, A. A. 2015. Unsupervised Visual Representation Learning by Context Prediction. In ICCV.

Ganin, Y.; and Lempitsky, V. 2015. Unsupervised Domain Adaptation by Backpropagation. In ICML.

Gao, J.; Wang, J.; Dai, S.; Li, L.-J.; and Nevatia, R. 2019. NOTE-RCNN: NOise Tolerant Ensemble RCNN for Semi-Supervised Object Detection. In ICCV.

Geiger, A.; Lenz, P.; Stiller, C.; and Urtasun, R. 2013. Vision meets Robotics: The KITTI Dataset. IJRR.

Gidaris, S.; Singh, P.; and Komodakis, N. 2018. Unsupervised Representation Learning by Predicting Image Rotations. In ICLR.

He, L.; Liao, X.; Liu, W.; Liu, X.; Cheng, P.; and Mei, T. 2020. FastReID: A Pytorch Toolbox for General Instance Re-identification. arXiv preprint arXiv:2006.02631.

He, Z.; and Zhang, L. 2019. Multi-adversarial Faster-RCNN for Unrestricted Object Detection. In ICCV.

Inoue, N.; Furuta, R.; Yamasaki, T.; and Aizawa, K. 2018. Cross-Domain Weakly-Supervised Object Detection through Progressive Domain Adaptation. In CVPR.

Isobe, T.; Jia, X.; Chen, S.; He, J.; Shi, Y.; Liu, J.; Lu, H.; and Wang, S. 2021. Multi-Target Domain Adaptation with Collaborative Consistency Learning. In CVPR.

Jeong, J.; Lee, S.; Kim, J.; and Kwak, N. 2019. Consistency-based Semi-supervised Learning for Object Detection. In NeurIPS.

Jiang, L.; Meng, D.; Zhao, Q.; Shan, S.; and Hauptmann, A. G. 2015. Self-paced Curriculum Learning. In AAAI.

Khodabandeh, M.; Vahdat, A.; Ranjbar, M.; and Macready, W. G. 2019. A Robust Learning Approach to Domain Adaptive Object Detection. In ICCV.

Kim, S.; Choi, J.; Kim, T.; and Kim, C. 2019. Self-Training and Adversarial Background Regularization for Unsupervised Domain Adaptive One-Stage Object Detection. In ICCV.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In ICLR.

Li, B.; Ren, W.; Fu, D.; Tao, D.; Feng, D.; Zeng, W.; and Wang, Z. 2019. Benchmarking single-image dehazing and beyond. TIP, 492–505.

Li, X.; Chen, W.; Xie, D.; Yang, S.; Yuan, P.; Pu, S.; and Zhuang, Y. 2021a. A Free Lunch for Unsupervised Domain Adaptive Object Detection without Source Data. In AAAI.

Li, X.; Chen, W.; Xie, D.; Yang, S.; Yuan, P.; Pu, S.; and Zhuang, Y. 2021b. A Free Lunch for Unsupervised Domain Adaptive Object Detection without Source Data. In AAAI.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollar, P. 2017. Focal Loss for Dense Object Detection. In ICCV.

Lin, T.-Y.; Maire, M.; Belongie, S.; Bourdev, L.; Girshick, R.; Hays, J.; Perona, P.; Ramanan, D.; Zitnick, C. L.; and Dollar, P. 2014. Microsoft COCO: Common Objects in Context. In ECCV.

Matthew, J.-R.; Charles, B.; Rounak, M.; Nittur, S. S.; Karl, R.; and Ram, V. 2017. Driving in the Matrix: Can virtual worlds replace human-generated annotations for real world tasks? In ICRA.

Munir, M. A.; Khan, M. H.; Sarfraz, M. S.; and Ali, M. 2021. Synergizing between Self-Training and Adversarial Learning for Domain Adaptive Object Detection. In NeurIPS.

Nada, H.; Sindagi, V. A.; Zhang, H.; and Patel, V. M. 2018. Pushing the Limits of Unconstrained Face Detection: a Challenge Dataset and Baseline Results. arXiv:1804.10275.

Redmon, J.; and Farhadi, A. 2018. YOLOv3: An Incremental Improvement. arXiv:1804.02767.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In NeurIPS.

Rezaeianaran, F.; Shetty, R.; Aljundi, R.; Reino, D. O.; Zhang, S.; and Schiele, B. 2021. Seeking Similarities over Differences: Similarity-based Domain Alignment for Adaptive Object Detection. In ICCV.

Roy, S.; Krivosheev, E.; Zhong, Z.; Sebe, N.; and Ricci, E. 2021. Curriculum Graph Co-Teaching for Multi-Target Domain Adaptation. In CVPR.

RoyChowdhury, A.; Chakrabarty, P.; Singh, A.; Jin, S.; Jiang, H.; Cao, L.; and Learned-Miller, E. 2019. Automatic adaptation of object detectors to new domains using self-training. In CVPR.

Sakaridis, C.; Dai, D.; and Gool, L. V. 2018. Semantic Foggy Scene Understanding with Synthetic Data. IJCV, 973–992.

Tang, Z.; Naphade, M.; Liu, M.-Y.; Yang, X.; Birchfield, S.; Wang, S.; Kumar, R.; Anastasiu, D.; and Hwang, J.-N. 2019. CityFlow: A City-Scale Benchmark for Multi-Target Multi-Camera Vehicle Tracking and Re-Identification. In CVPR.

Verma, V.; Lamb, A.; Kannala, J.; and Bengio, Y. 2019. Interpolation Consistency Training for Semi-Supervised Learning. In IJCAI.

Wang, Y.; Zhang, R.; Zhang, S.; Li, M.; Xia, Y.; Zhang, X.; and Liu, S. 2021. Domain-Specific Suppression for Adaptive Object Detection. In CVPR.

Wu, A.; Liu, R.; Han, Y.; Zhu, L.; and Yang, Y. 2021. Vector-Decomposed Disentanglement for Domain-Invariant Object Detection. In ICCV.

Xin, C.; and K, T. J. 2019. Fast Visual Object Tracking with Rotated Bounding Boxes. In ICCV Workshops.

Yang, J.; Shi, S.; Wang, Z.; Li, H.; and Qi, X. 2021a. ST3D: Self-training for Unsupervised Domain Adaptation on 3D Object Detection. In CVPR.

Yang, Q.; Wei, X.; Wang, B.; Hua, X.-S.; and Zhang, L. 2021b. Interactive Self-Training with Mean Teachers for Semi-supervised Object Detection. In CVPR.

Yang, T.; Zhang, X.; Li, Z.; Zhang, W.; and Sun, J. 2020. Fix-Match: Simplifying Semi-Supervised Learning with Consistency and Confidence. In NeurIPS.

Yu, F.; Chen, H.; Wang, X.; Xian, W.; Chen, Y.; Liu, F.; Madhavan, V.; and Darrell, T. 2020. BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning. In CVPR.

Zhang, R.; Isola, P.; and Efros, A. A. 2016. Colorful Image Colorization. In ECCV.

Zhang, Y.; Wang, Z.; and Mao, Y. 2021. RPN Prototype Alignment For Domain Adaptive Object Detector. In ICCV.

Zhang, Z. 1998. Determining the Epipolar Geometry and its Uncertainty: A Review. IJCV, 161–198.

Zhao, Q.; Sheng, T.; Wang, Y.; Tang, Z.; Chen, Y.; Cai, L.; and Ling, H. 2019. M2Det: A Single-Shot Object Detector based on Multi-Level Feature Pyramid Network. In AAAI.

Zheng, Y.; Huang, D.; Liu, S.; and Wang, Y. 2020a. Cross-domain Object Detection through Coarse-to-Fine Feature Adaptation. In CVPR.

Zheng, Z.; Ruan, T.; Wei, Y.; Yang, Y.; and Mei, T. 2020b. VehicleNet: Learning Robust Visual Representation for Vehicle Reidentification. arXiv:2004.06305.

Zhuang, C.; Han, X.; Huang, W.; and Scott, M. R. 2020. iFAN: Image-Instance Full Alignment Networks for Adaptive Object Detection. In AAAI.