A Free Lunch for Unsupervised Domain Adaptive Object Detection without Source Data

Xianfeng Li, [1*] **Weijie Chen,** [32] **Di Xie,** [2] **Shicai Yang,** [2]
Peng Yuan, [2] **Shiliang Pu,** [2†] **Yueting Zhuang**[3]

[1] South China University of Technology [2] Hikvision Research Institute [3] Zhejiang University
lockonlxf@163.com, {chenweijie5, xiedi, yangshicai, yuanpeng7, pushiliang.hri}@hikvision.com, yzhuang@zju.edu.cn

Abstract

Unsupervised domain adaptation (UDA) assumes that source and target domain data are freely available and usually trained together to reduce the domain gap. However, considering the data privacy and the inefficiency of data transmission, it is impractical in real scenarios. Hence, it draws our eyes to optimize the network in the target domain without accessing labeled source data. To explore this direction in object detection, for the first time, we propose a *source data-free domain adaptive object detection* (SFOD) framework via modeling it into a problem of learning with noisy labels. Generally, a straightforward method is to leverage the pre-trained network from the source domain to generate the pseudo labels for target domain optimization. However, it is difficult to evaluate the quality of pseudo labels since no labels are available in target domain. In this paper, *self-entropy descent* (SED) is a metric proposed to search an appropriate confidence threshold for reliable pseudo label generation without using any handcrafted labels. Nonetheless, completely clean labels are still unattainable. After a thorough experimental analysis, false negatives are found to dominate in the generated noisy labels. Undoubtedly, false negatives mining is helpful for performance improvement, and we ease it to false negatives simulation through data augmentation like Mosaic. Extensive experiments conducted in four representative adaptation tasks have demonstrated that the proposed framework can easily achieve state-of-the-art performance. From another view, it also reminds the UDA community that the labeled source data are not fully exploited in the existing methods.

Introduction

Deep convolutional neural networks have significantly improved object detection performance (Ren et al. 2015; Redmon et al. 2016; Liu et al. 2016) but rely on large quantities of high-quality manual annotated training data. This limits the ability to generalize when facing new environments or data distributions where the object appearance, scene type, illumination, background, or weather condition are various. It attracts us to study how to transfer the pre-trained model from a label-rich source domain to an unlabeled target domain without supervision.

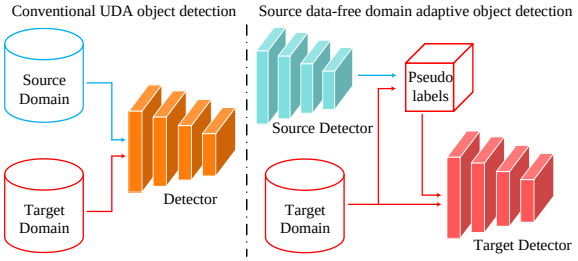

Figure 1: The comparison between conventional UDA object detection and our proposed source data-free domain adaptive object detection.

Various unsupervised domain adaptive methods had been proposed to tackle this problem, whether using domain-invariant features for alignment (Chen et al. 2018; Saito et al. 2019; He and Zhang 2020; Xu et al. 2020), or narrowing the distribution of the domain in the image space (Liu, Breuel, and Kautz 2017; Hoffman et al. 2018; Hsu et al. 2020), or using pseudo label techniques by measuring the similarity between source and target domain samples (C.Chen et al. 2019; Q.Wang and T.Breckon 2020). These methods align the distributions of source and target domain, assuming that the data distribution of labeled source domain and unlabeled target domain is related but different (Sugiyama and Storkey 2007) and needing access freely to both source and target domain samples. However, this assumption will encounter challenges in practical application, such as data privacy and impractical data transmission.

At present, some classification methods (Li et al. 2020; Kim, Hong, and Cho 2020; Peng et al. 2020) about source data-free have made good progress, but there is still a blank in the source data-free unsupervised domain adaptive object detection. This paper proposes a simple yet effective approach to the above problems named *source data-free domain adaptive object detection* (SFOD), which decouples the domain adaptation process by leveraging a pre-trained source model. The key idea is to train the target model with reliable pseudo labels of target samples in a self-learning manner. The natural question is how to evaluate the quality of pseudo labels for object detection and learn with noisy labels. In the classification task, the total number of samples is fixed. Even if only a few reliable pseudo labels are

*Internship at Hikvision Research Institute.
†Shiliang Pu is the Corresponding Author

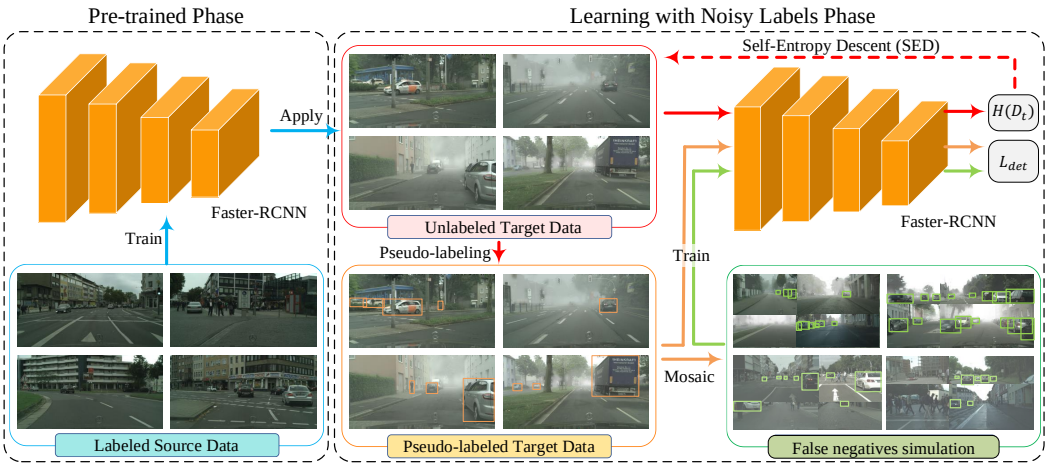

Figure 2: The pipeline of the proposed source data-free domain adaptative object detection (SFOD). The given supervision signals are only provided by the pre-trained model from source domain during adaptation.

used, more reliable pseudo labels can be mined to achieve pseudo labels refinement (Kim, Hong, and Cho 2020). However, it is more challenging in the object detection task, since the negative samples are countless and various, and lots of hard positive samples are difficult to box out and mixed with negative samples. Only relying on a small number of reliable samples cannot achieve good performance. A straightforward method is to directly filter out the bounding boxes into positive parts and negative parts according to an appropriate confidence threshold. Although there unavoidably are some false positive and false negative samples (namely noisy labels in object detection), the target model can still be optimized following the 'wisdom of the crowd' principle. However, an appropriate confidence threshold is difficult to search since no metric is available for supervision. It would harm the network performance if confidence threshold is too high or too low due to the messier noisy labels. This inspires us to search for an appropriate confidence threshold for pseudo label generation to make a trade-off between the positive effect brought by true positives and the negative effect brought by false positives and false negatives.

In this paper, a metric named *self-entropy descent* (SED) is proposed to search the confidence threshold. As is known, prediction uncertainty can be quantified as self-entropy, i.e., $H(x) = -\sum p(x)\log(p(x))$. The lower the self-entropy the more confident the prediction. Here we search the confidence threshold from the higher score to the lower score. Meanwhile, we use the generated pseudo label to fine-tune the pretrained model and then evaluate the self-entropy of the dataset after training (namely mean self-entropy). Note that the noisier the labels, the more difficult to fit the labels. Therefore, as the confidence threshold decreases, when the mean self-entropy descends and hits the first local minimum, we select it as an appropriate confidence threshold for reliable pseudo label generation. We design a toy experiment to prove the reasonability of this solution. Nonetheless, we have to admit the generated pseudo labels are still unavoidably noisy. Specifically, there exist false positives and false negatives in the generated pseudo labels. Through a thorough experimental analysis in the publicly-released datasets, false negatives are found to dominate in the noisy labels, such as small and obscured objects. Hence, to alleviate the effects from false negatives, false negatives mining is proposed to solve this problem. And we ease this solution to false negatives simulation via data augmentation like Mosaic (Bochkovskiy, Wang, and Liao 2020), since it can exploit the easy positive samples to simulate false negative samples. We believe more label denoising techniques can further boost the performance, and we leave this as our future work.

The main contributions of this work are summarized as follows. (i) To the best of our knowledge, this is the first work on source data-free unsupervised domain adaptative object detection. (ii) We innovatively model the source data-free UDA into a problem of learning with noisy labels and make it solvable. (iii) Our framework can achieve delectable performance without using source data and surpass most of the other source data based UDA methods. It implies the UDA community that the labeled source data are not fully exploited in the existing UDA methods in object detection.

Related Works

Domain Adaptive Object Detection

The proposal of *Domain Adaptive Faster R-CNN* (Chen et al. 2018) has made progress in the challenging unsupervised domain adaptative object detection task, which aligns both the image and instance levels in a domain adversarial manner. After that, the following works *Strong-Weak Domain Adaptive Faster R-CNN* (Saito et al. 2019), *Region-level Alignment* (Zhu et al. 2019), *Categorical Regularization Domain Adaptive Object Detection* (Xu et al. 2020), *Asymmetric Tri-way Faster-RCNN* (He and Zhang 2020), and style transfer based method (Hsu et al. 2020) were proposed one after another to push this direction forward. However, the existing methods require both labeled source data and unlabeled target data, while our proposed source data-free one is more practical in real scenarios.

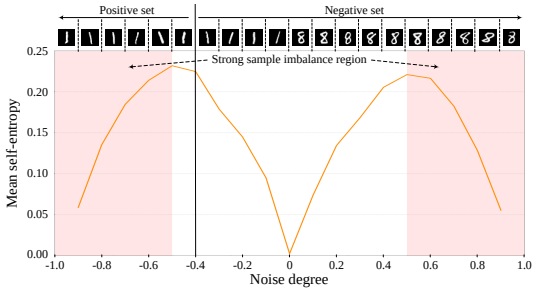

Figure 3: A toy example to capture the relation between noisy labels and mean self-entropy. Noise degree denotes the ratio of positive samples mixed into negative set (-) and the ratio of negative samples mixed into positive set (+). Two local minimums appearing in two ends of the curve are resulted from strong sample imbalance.

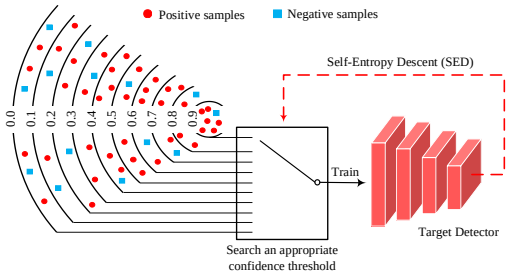

Figure 4: An appropriate confidence threshold to split positive and negative objects is searched from the higher score to the lower score with the metric of SED.

Domain Adaptation without Source Data

Considering data privacy and data transmission, some source data-free domain adaptative classification approaches have been proposed (Li et al. 2020; Kim, Hong, and Cho 2020). However, there is still a blank in source data-free unsupervised domain adaptive object detection.

Learning with Noisy Labels

The current research about learning with noisy labels still focuses on relatively simple classification tasks. Earlier work in this field used an instance-independent noise model, where each class was confused with other classes that were independent of the content of the instance (Mnih and Hinton 2012; Natarajan et al. 2013; Patrini et al. 2017). Recently, some methods focus on specific examples of label noise prediction (Vahdat 2017; Veit et al. 2017; Ren et al. 2018; Jiang et al. 2018). However, the noisy labels setting in these researches is ideal, where the noisy labels and true labels are manually set and identically distributed. While in our work, the noisy labels are not manually set and not identically distributed, where the noisy labels are both hard positive and negative objects. Moreover, there are few methods about learning with noisy labels designed for object detection. (Khodabandeh et al. 2019) is the one easing the object detection problem into image classification, but it cannot solve the situation when the objects are hard to box out.

Source free Domain Adaptive Object Detection

The assumption of unsupervised domain adaptive object detection is that the data of labeled source domain $D_s = \left\{\left(x_s^i, y_s^i\right)\right\}_{i=1}^{N_s}$ and unlabeled target domain $D_t = \left\{x_t^i\right\}_{i=1}^{N_t}$ are available freely in training step to minimize the discrepancy between them. Unlike this learning paradigm, source data-free UDA aims to optimize the network only through the unlabeled target domain $D_t = \left\{x_t^i\right\}_{i=1}^{N_t}$. The only supervision signal is given by the pre-trained model θ_s from source domain instead of directly using source domain data.

Pseudo Labels Optimization via SED

A toy example: How to evaluate the quality of pseudo labels? In this section, a toy example on two categories of MNIST (LeCun, Cortes, and Burges 1998) dataset representing the positive and negative samples, called MNIST-2, is presented. To study how to evaluate the quality of pseudo labels, we build different datasets based on MNIST-2 through mixing different proportions of positive samples into the negative part or mixing different proportions of negative samples into the positive part and use LeNet (Lecun et al. 1998) to train these datasets. For simplicity, the mixing proportion is also named as noise degree. And a notion named mean self-entropy is introduced to capture the uncertainty of the prediction of the entire dataset after training which can be formulated as follows:

$$H\left(D_t\right) = -\frac{1}{N_t}\sum_{i}^{N_t}\left(\frac{1}{n_c}\sum_{c}^{n_c} p_c\left(x_t^i\right)\log\left(p_c\left(x_t^i\right)\right)\right) \quad (1)$$

where n_c refer to the class number, and $p_c\left(x_t^i\right)$ denotes the prediction probability of class c, respectively.

Unsurprisingly, as shown in Figure 3, the noise degree is positively correlated with mean self-entropy. The noisier the labels, the more difficult to fit the labels, which leads to larger mean self-entropy. Note that two local minimums in two ends of the mean self-entropy curve are resulted from a strong sample imbalance. Ideally, the cleanest label assignment will lead to the lowest mean self-entropy. Considering both situations, it indicates a reliable label assignment when mean self-entropy descends and hits the local minimum.

Self-entropy descent: how to generate reliable pseudo labels in object detection? When it comes to object detection, the negative samples are countless and various. Based on the clue in the above section, we search an appropriate confidence threshold from the higher score to the lower score to split positive and negative samples for training and stop when the mean self-entropy descends and hits the FIRST local minimum. We name it as *Self-entropy descent*.

The unlabeled target domain $D_t = \left\{x_t^i\right\}_{i=1}^{N_t}$ and the source domain's pre-trained model θ_s are available freely. So pseudo labels $y\left(x_t\right)$ and the corresponding confidence $p\left(x_t\right)$ can be obtained as follows:

$$\left\{y\left(x_t^i\right), p\left(x_t^i\right)\right\}_{i=1}^{N_t} = \left\{F\left(x_t^i \mid h, \theta_s\right)\right\}_{i=1}^{N_t} \quad (2)$$

where h is a confidence threshold for pseudo label generation, and F represents Faster-RCNN (Ren et al. 2015) de-

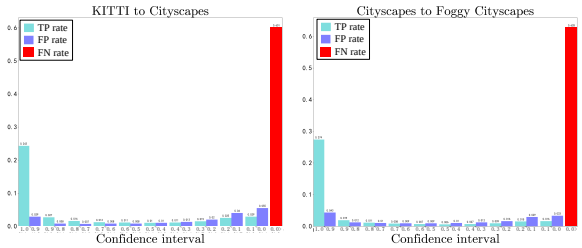

Figure 5: In two cross-domain datasets, KITTI to Cityscapes and Cityscapes to Foggy Cityscapes, the ratio of true positives and false positives to the entire ground truth are counted in different confidence intervals. The confidence of target domain data is directly predicted by the pre-trained model from source domain. False negatives (<0.0), which are difficult to box out even when the confidence threshold is set to zero, are found to dominate in the noisy labels.

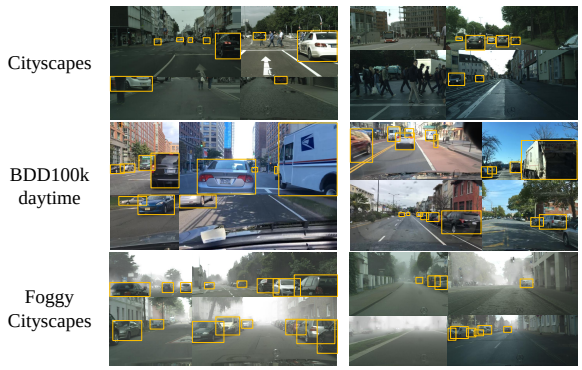

Figure 6: Mosaic visualization with pseudo labels.

tector. Faster-RCNN is the first anchor-based object detection method, where the detector has an encoder network as a feature extractor, a *Region Proposal Network* (RPN) and *Region of Interest* (ROI) classifier.

Specifically, the confidence $p\left(x_t\right)$ is the output of softmax in the classification branch. And the pseudo label $y\left(x_t\right)$ is determined by the argmax of foreground class probability. If this score is greater than the given confidence threshold h, the corresponding box will be assigned as the class label with the max score; otherwise, it will be assigned as the background class. To train the target domain data with pseudo labels, the loss function is formulated as:

$$L_{det} = L_{rpn} + L_{cls} + L_{reg} \tag{3}$$

where L_{rpn}, L_{cls}, and L_{reg} denotes the region proposal loss, region classification loss and the bounding-box regression loss. As for region proposal and bounding boxes regression, we directly use the bounding boxes predicted by the pre-trained model from the source domain as ground-truth for training. As claimed by (Borji and Iranmanesh 2019), the location error is much weaker than classification error in object detection task.

After fine-tuning the pre-trained model with the pseudo-labels generated by a given confidence threshold, we use the updated model θ_t to evaluate the mean self-entropy $H\left(D_t\right)$ of the target datasets.

$$\theta_t = Train(\{x_t^i, y_t^i\}|\theta_s) \tag{4}$$

$$\left\{-, \hat{p}\left(x_t^i\right)\right\}_{i=1}^{N_t} = \left\{F\left(x_t^i \mid h, \theta_t\right)\right\}_{i=1}^{N_t} \tag{5}$$

$$H\left(D_t\right) = -\frac{1}{N_t}\sum_{i}^{N_t}\left(\frac{1}{n_c}\sum_{c}^{n_c}\hat{p}_c\left(x_t^i\right)\log\left(\hat{p}_c\left(x_t^i\right)\right)\right) \tag{6}$$

According to the SED policy, we search the confidence threshold from the higher to the lower score, and early stop when $H\left(D_t\right)$ descends and hit the first local minimum.

$$h_{optimal} = \arg\min_{h} H\left(D_t\right) \tag{7}$$

False Negatives Simulation

Although we search for an appropriate confidence threshold via SED, we have to admit the generated pseudo labels are still noisy. Label denoise techniques can be applied to clean the labels and boost performance. In an object detection task, the noisy labels behave as false positives and false negatives. We count the true positives and false positives in each confidence interval in several publicly-released datasets. As shown in Figure 5, false positives only account for a relatively small proportion. And surprisingly, more than 50% positive samples are difficult to box out even though we set the confidence threshold close to zero, which behave as false negatives during training. Therefore, in this paper, we mainly focus on false negatives mining for labels denoising.

Through visualization, most false negatives are small and obscured objects mixed with true negatives, which are very difficult to mine back into the positive part. The domain gap between the source domain and target domain increases the difficulty of detecting hard examples. Hence, we ease this solution to false negatives simulation by exploiting true positives. Data augmentation is a good way to augment the detected positives into hard ones to simulate the small and obscured objects. It can suppress the negative effects of false negatives. In this work, Mosaic augmentation (Bochkovskiy, Wang, and Liao 2020) is selected for false negatives simulation since it can generate small-scale and blocked objects by exploiting true positives while not harming the true negatives. Mosaic is the improvement of CutMix (Yun et al. 2019) via mixing four training images, which allows the detection of objects outside their normal context. The two main steps in Mosaic are random scaling and random cutting. The hard objects with different scales can be simulated by using the simple objects that have been detected in the target domain via using random scaling. Meanwhile, the blocked objects with the only visible part of the structure can be simulated to a certain extent by random cutting. Mosaic data $\{(\tilde{x}, \tilde{y})\}$ can be formulated by the target domain data $\{(x_A, y_A), (x_B, y_B), (x_C, y_C), (x_D, y_D)\}$ as follows:

$$\tilde{x} = \begin{bmatrix} M_A \odot s\left(x_A\right) & M_B \odot s\left(x_B\right) \\ M_C \odot s\left(x_C\right) & M_D \odot s\left(x_D\right) \end{bmatrix} \in R^{W \times H} \tag{8}$$

$$\begin{aligned} \tilde{y} = &(\lambda, \lambda) \cdot s\left(y_A\right) + (\lambda, 1-\lambda) \cdot \left(s\left(y_B\right) + v\right) \\ &+(1-\lambda, \lambda) \cdot \left(s\left(y_c\right) + u\right) \\ &+(1-\lambda, 1-\lambda) \cdot \left(s\left(y_D\right) + (u, v)\right) \end{aligned} \tag{9}$$

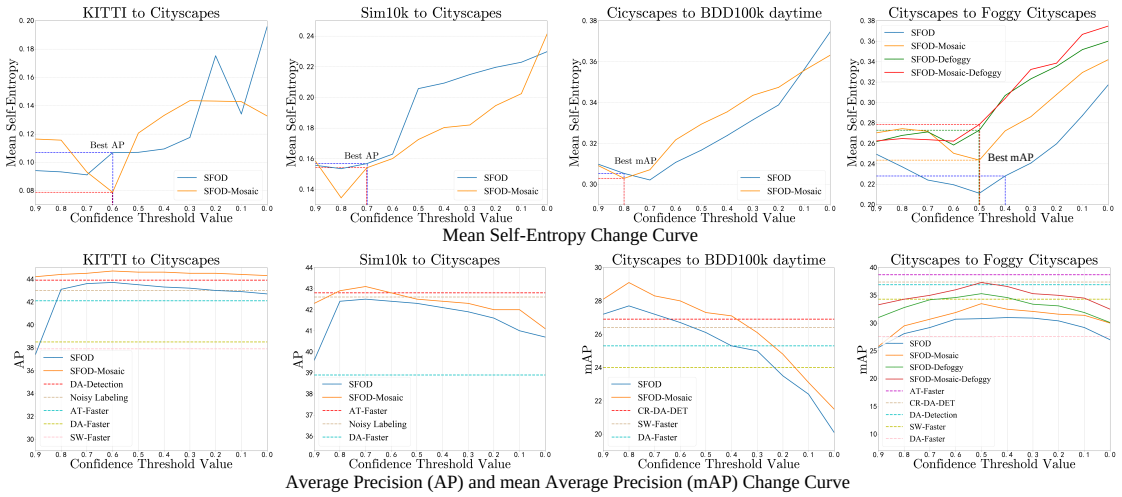

Figure 7: The curves of mean self-entropy and the corresponding AP or mAP vary with confidence threshold in four adaptation tasks. It can nearly search the best mAP via SED.

Methods	AP of Car
Source only	36.4
DA-Faster (Chen et al. 2018)	38.5
SW-Faster (Saito et al. 2019)	37.9
MAF (Z.He and L.Zhang 2019)	41.0
AT-Faster (He and Zhang 2020)	42.1
Noise Labeling (Khodabandeh et al. 2019)	43.0
DA-Detection (Hsu et al. 2020)	43.9
SFOD (SED)	43.6
SFOD-Mosaic (SED)	44.6
SFOD (Ideal)	43.7
SFOD-Mosaic (Ideal)	44.6
Oracle	58.5

Table 1: Results of adaptation to a new sense, i.e., from KITTI dataset to Cityscapes dataset.

Methods	AP of Car
Source only	33.7
DA-Faster (Chen et al. 2018)	38.5
MAF (Z.He and L.Zhang 2019)	41.1
AT-Faster (He and Zhang 2020)	42.1
Noise Labelling (Khodabandeh et al. 2019)	43.0
SFOD (SED)	42.3
SFOD-Mosaic (SED)	42.9
SFOD (Ideal)	42.5
SFOD-Mosaic (Ideal)	43.1
Oracle	58.5

Table 2: Results of adaptation from synthetic to real images, i.e.,from Sim10k dataset to Cityscapes dataset.

where W and H represent the size of training images, $\{M_A, M_B, M_C, M_D\} \in \{0,1\}^{s(W)\times s(H)}$ denotes a group of binary masks, (u, v) is the 2D translation, $s(\cdot)$ and λ represent random scaling function and random cutting factor. Figure 6 displays some Mosaic images. We believe more effective false negatives mining or false negatives simulation methods can bring further performance boost.

False negatives simulation is adopted with SED to search an appropriate confidence threshold for pseudo label generation. The entire pipeline of SFOD is shown in Figure 2.

Experiments

Experimental Setup

Datasets Five public datasets are utilized in our experiments. (1) **KITTI** (Geiger, Lenz, and Urtasun 2012) is a popular dataset for autonomous driving, which are manually collected in several different scenes in a city with 7,481 labeled images for training. (2) **Sim10k** (Johnson-Roberson et al. 2017) simulates different scenes, such as different times or weather, from a computer game *Grand Theft Auto V* (GTA V) with 10k images. (3) **Cityscapes** (Cordts et al. 2016) focuses on the high variability of outdoor street scenes from different cities. We transform the instance segmentation annotations of 2,975 training images and 500 validation images into bounding boxes for our experiments. (4) **BDD100k** (Yu et al. 2018) includes 100k images with six types of weather, six different scenes, and three categories for the time of day. We extract a subset labeled as daytime, including 36,728 training and 5,258 validation images. (5) **Foggy Cityscapes** (Sakaridis, Dai, and Gool 2018) simulates the foggy weather using city images from Cityscapes with three foggy weather levels and inherit annotations of Cityscapes.

Implementation Details For a fair comparison with existing approaches, we follow the same experimental setting as (Chen et al. 2018; Xu et al. 2020). The short size of all training and testing images are resized to a length of 600 pixels. We use the pre-trained weights of VGG-16 (Simonyan

Methods	truck	car	rider	person	train	motor	bicycle	bus	mAP
Source only	14.0	40.7	24.4	22.4	-	14.5	20.5	16.1	21.8
DA-Faster (Chen et al. 2018)	14.3	44.6	26.5	29.4	-	15.8	20.6	16.8	24.0
SW-Faster (Saito et al. 2019)	15.2	45.7	29.5	30.2	-	17.1	21.2	18.4	25.3
CR-DA-DET (Xu et al. 2020)	19.5	46.3	31.3	31.4	-	17.3	23.8	18.9	26.9
SFOD (SED)	20.4	48.8	32.4	31.0	-	15.0	24.3	21.3	27.6
SFOD-Mosaic (SED)	20.6	50.4	32.6	32.4	-	18.9	25.0	23.4	29.0
SFOD (Ideal)	20.0	46.8	32.1	31.5	-	16.3	25.1	21.8	27.7
SFOD-Mosaic (Ideal)	20.6	50.4	32.6	32.4	-	18.9	25.0	23.4	29.0
Oracle	53.4	53.5	42.8	41.9	-	37.3	38.8	58.1	47.1

Table 3: Results of adaptation to a large-scale dataset, i.e.,from Cityscapes dataset to BDD100k daytime dataset.

and Zisserman 2015) on ImageNet (Deng et al. 2009) as the backbone of the Faster-RCNN framework. The detector is trained with *Stochastic Gradient Descent* (SGD) with a learning rate of 0.001. The batch size is set to 1. Source domain data are only used in the pre-trained step.

Comparison Results

Our experiments are carried out in four adaptation tasks. Figure 7 shows the curves of detection precision and mean self-entropy under different confidence thresholds for pseudo label generation. "Source only" and "Oracle" are both tested in target domain validation set, but trained with labeled source domain training set and labeled target domain training set, respectively.

Adaptation to A New Sense Different camera setups (e.g., angle, resolution, quality, and type) widely exist in the real world, which can cause the domain shift. In this experiment, we take the adaptation to a new sense task between two real datasets. The KITTI and Cityscapes datasets are used as source and target domains, respectively. We implement our SFOD, DA-Faster (Chen et al. 2018), SW-Faster (Saito et al. 2019), Noise Labeling (Khodabandeh et al. 2019), DA-Detection (Hsu et al. 2020), and AT-Faster (He and Zhang 2020) in this task. In Table 1, the *average precision* (AP) on the car category, the only common object, is compared. When SED is used alone, although the ideal confidence threshold searched by the labeled target validation set is not found, the AP is very close to the ideal one, and our method has surpassed many existing methods in terms of car detection accuracy. When Mosaic is further used, the AP can be increased from 43.6% to 44.6% and exceeds DA-Detection (Hsu et al. 2020) by 0.7%. We can see that false negatives simulation can ease the negative effect brought by the false negative noisy labels.

Adaptation from Synthetic to Real Images Another domain adaptation scenario is from synthetic data to the real world. Due to the lack of annotated training data to autonomous driving, synthetic data offers an alternative. Thus, the source domain is the Sim10k, and the target domain is the Cityscapes. In this task, we only evaluate the performance in annotated cars for which is the only object category in both Sim10k and Cityscapes. In Table 2, compared with DA-Faster (Chen et al. 2018), Noise Labeling (Khodabandeh et al. 2019), and AT-Faster (He and Zhang 2020), our source data-free method can achieve superior or comparable results. However, our source data-free setting is more challenging than the existing source data-based methods.

Adaptation to Large-Scale Dataset Currently, collecting large amounts of image data is not difficult, but labeling those data is still the main problem for supervised learning methods. In this experiment, we use Cityscapes as a smaller source domain dataset, BDD100k containing distinct attributes as a large unlabeled target domain dataset. Since there is only daytime data in Cityscapes, we select the labeled daytime data in the three-time periods of BDD100k as the target domain. We evaluate the *mean average precision* (mAP) of detection results on seven categories in both datasets. As we can see from the baseline and oracle results in Table 3, resolving such a domain divergence between a source domain and a target domain is so complicated that only a handful of approaches (e.g., DA-Faster (Chen et al. 2018), SW-Faster (Saito et al. 2019), and CR-DA-DET (Xu et al. 2020)) challenge this adaptation task, let alone source data-free. Even with such a wide range of the domain gap, it is surprising to see in Figure 7 that the state-of-the-art methods are improved over a wide range of confidence thresholds. Especially when we use SED or SED+Mosaic, we can improve the mAP from 26.9% of CR-DA-DET (Xu et al. 2020) to 27.6% and 29.0%.

Adaptation from Normal to Foggy Weather In real-world applications, object detectors may be used with different weather conditions. It is hard to collect and label a large number of data from every weather condition. To study the changing environment adaptation from normal weather to a foggy condition, Cityscapes and Foggy Cityscapes are used as the source domain and the target domain, respectively. The comparisons between our SFOD and other UDA object detection methods (i.e., DA-Faster (Chen et al. 2018), SW-Faster (Saito et al. 2019), DA-Detection (Hsu et al. 2020), CR-DA-DET (Xu et al. 2020), and AT-Faster (He and Zhang 2020)) are presented on eight common categories in Table 4. Compared to 22.3% mAP of the baseline, even using pseudo labels with label noise trained by SED and Mosaic can still be improved to 33.5%. However, there is still a certain gap to achieve the performance of the traditional UDA object de-

Faster-RCNN (source only) SED (ours) SED + Mosaic (ours)

Figure 8: Qualitative results. Top: KITTI to Cityscapes. Bottom: Cityscapes to Foggy Cityscapes. Red, green and blue boxes denote true positives, false negatives and false positives.

Methods	defoggy	truck	car	rider	person	train	motor	bicycle	bus	mAP
Source only	×	11.6	38.7	31.4	23.6	9.4	17.3	27.4	19.0	22.3
DA-Faster (Chen et al. 2018)	×	19.5	43.5	36.5	28.7	12.6	24.8	29.1	33.1	28.5
MAF (Z.He and L.Zhang 2019)	×	23.8	43.9	39.5	28.2	33.3	29.2	33.9	39.9	34.0
SW-Faster (Saito et al. 2019)	×	23.7	47.3	42.2	32.3	27.8	28.3	35.4	41.3	34.8
DA-Detection (Hsu et al. 2020)	√	24.3	54.4	45.5	36.0	25.8	29.1	35.9	44.1	36.9
CR-DA-DET (Xu et al. 2020)	×	27.2	49.2	43.8	32.9	36.4	30.3	34.6	45.1	37.4
AT-Faster (He and Zhang 2020)	×	23.7	50.0	47.0	34.6	38.7	33.4	38.8	43.3	38.7
SFOD (SED)	×	21.7	44.0	40.4	32.6	11.8	25.3	34.5	34.3	30.6
SFOD-Mosaic (SED)	×	25.5	44.5	40.7	33.2	22.2	28.4	34.1	39.0	33.5
SFOD (Ideal)	×	22.3	44.0	38.2	31.4	15.1	25.7	34.6	36.8	31.0
SFOD-Mosaic (Ideal)	×	25.5	44.5	40.7	33.2	22.2	28.4	34.1	39.0	33.5
SFOD-Defoggy (SED)	√	28.4	50.9	41.6	32.2	15.9	28.1	36.0	40.1	34.2
SFOD-Mosaic-Defoggy (SED)	√	27.9	51.7	44.7	33.2	21.3	28.6	37.3	45.9	36.3
SFOD-Defoggy (Ideal)	√	26.2	50.6	41.8	32.5	24.4	28.7	36.1	40.5	35.1
SFOD-Masoic-Defoggy (Ideal)	√	30.4	51.9	44.4	34.1	25.7	30.3	37.2	41.8	37.0
Oracle	×	38.1	49.8	53.1	33.1	37.4	41.1	57.4	48.2	44.8

Table 4: Results of adaptation from normal to foggy dataset, i.e.,from Cityscapes dataset to Foggy Cityscapes dataset.

tection methods. As a further discussion, we used the same defogging method like DA-Detection (Hsu et al. 2020) to improve the image quality of the target domain, and then studied the performance of SFOD under this condition. As we can see from Table 4, SFOD performance has been improved by approximately 3% after defogging. Based on the above phenomenon, it can be concluded that the fog aggravates the label noise in pseudo labels, thus affecting the detection performance.

Discussion and Analysis

In SFOD, the training process with pseudo labels of target domain data obtained by using the source domain's pre-trained model will be disturbed because of noisy labels. Some object detection results are shown in Figure 8, whether using SED directly to search an appropriate confidence threshold for pseudo label generation or further combining with false negatives simulation, the negative effects brought by noisy labels can be well suppressed so that more objects can be detected. Our proposed SFOD achieves comparable even superior results to the existing source data based UDA methods, which means the source domain data is actually not fully exploited in the existing methods.

Conclusion

In this paper, we propose a new learning paradigm for unsupervised domain adaptive object detection named SFOD. The challenge lies in only utilizing a pre-trained model from the source domain instead of directly using source data to provide supervision signals. We make it solvable from the view of learning with noisy labels. Although our method even surpasses many source data-based methods, we have to admit that to completely remove noisy labels (false positives and false negatives) is still very difficult in an unsupervised way. This is a very critical problem in SFOD, and our

work is the first try in this direction and hopes to bring more inspirations to the UDA community.

References

Bochkovskiy, A.; Wang, C.-Y.; and Liao, H.-Y. M. 2020. YOLOv4: Optimal Speed and Accuracy of Object Detection. In *arXiv:2004.10934*.

Borji, A.; and Iranmanesh, S. M. 2019. Empirical Upper Bound in Object Detection and More. *arXiv:1911.12451* .

C.Chen; W.Xie; W.Huang; Y.Rong; X.Ding; Y.Huang; T.Xu; and J.Huang. 2019. Progressive Feature Alignment for Unsupervised Domain Adaptation. In *CVPR*.

Chen, Y.; Li, W.; Sakaridis, C.; Dai, D.; and Gool, L. V. 2018. 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*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*.

Geiger, A.; Lenz, P.; and Urtasun, R. 2012. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*.

He, Z.; and Zhang, L. 2020. Domain Adaptive Object Detection via Asymmetric Tri-way Faster-RCNN. In *ECCV*.

Hoffman, J.; Tzeng, E.; Park, T.; Zhu, J.-Y.; Isola, P.; Saenko, K.; Efros, A. A.; and Darrell, T. 2018. Cycada: Cycle-consistent adversarial domain adaptation. In *ICML*.

Hsu, H.-K.; Yao, C.-H.; Tsai, Y.-H.; Hung, W.-C.; Tseng, H.; Singh, M.; and Yang, M.-H. 2020. Progressive Domain Adaptation for Object Detection. In *WACV*.

Jiang, L.; Zhou, Z.; Leung, T.; Li, L.-J.; and Fei-Fei, L. 2018. Mentornet: Regularizing very deep neural networks on corrupted labels. In *ICML*.

Johnson-Roberson, M.; Barto, C.; Mehta, R.; Sridhar, S. N.; Rosaen, K.; and Vasudevan, R. 2017. Driving in the matrix: Can virtual worlds replace humangenerated annotations for real world tasks? In *ICRA*.

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

Kim, Y.; Hong, S.; and Cho, D. 2020. Domain Adaptation without Source Data. In *arXiv:2007.01524*.

Lecun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86(11): 2278–2324.

LeCun, Y.; Cortes, C.; and Burges, C. 1998. The MNIST database of handwritten digits.

Li, R.; Jiao, Q.; Cao, W.; Wong, H.-S.; and Wu, S. 2020. Model Adaptation: Unsupervised Domain Adaptation without Source Data. In *CVPR*.

Liu, M.-Y.; Breuel, T.; and Kautz, J. 2017. Unsupervised image-to-image translation networks. In *NIPS*.

Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; and Berg, A. C. 2016. SSD: Single Shot MultiBox Detector. In *ECCV*.

Mnih, V.; and Hinton, G. E. 2012. Learning to label aerial images from noisy data. In *ICML*.

Natarajan, N.; Dhillon, I. S.; Ravikumar, P. K.; and Tewari, A. 2013. Learning with noisy labels. In *Advances in neural information processing systems*.

Patrini, G.; Rozza, A.; Menon, A.; Nock, R.; and Qu, L. 2017. Making neural networks robust to label noise: A loss correction approach. In *CVPR*.

Peng, X.; Huang, Z.; Zhu, Y.; and Saenko, K. 2020. Federated Adversarial Domain Adaptation. In *ICLR*.

Q.Wang; and T.Breckon. 2020. Unsupervised domain adaptation via structured prediction based selective pseudo-labeling. In *AAAI*.

Redmon, J.; Divvala, S.; Girshick, R.; and Farhadi, A. 2016. You Only Look Once: Unified, Real-Time Object Detection. In *CVPR*.

Ren, M.; Zeng, W.; Yang, B.; and Urtasun, R. 2018. Learning to reweight examples for robust deep learning. In *ICML*.

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

Saito, K.; Ushiku, Y.; Harada, T.; and Saenko, K. 2019. Strong-Weak Distribution Alignment for Adaptive Object Detection. In *CVPR*.

Sakaridis, C.; Dai, D.; and Gool, L. V. 2018. Semantic foggy scene understanding with synthetic data. In *IJCV*.

Simonyan, K.; and Zisserman, A. 2015. Very deep convolutional networks for large-scale image recognition. In *ICLR*.

Sugiyama, M.; and Storkey, A. 2007. Mixture regression for covariate shift. In *NIPS*.

Vahdat, A. 2017. Toward robustness against label noise in training deep discriminative neural networks. In *NIPS*.

Veit, A.; Alldrin, N.; Chechik, G.; Krasin, I.; Gupta, A.; and Belongie, S. 2017. Learning from noisy large-scale datasets with minimal supervision. In *CVPR*.

Xu, C.; Zhao, X.; Jin, X.; and Wei, X. 2020. Exploring Categorical Regularization for Domain Adaptive Object Detection. In *CVPR*.

Yu, F.; Xian, W.; Chen, Y.; Liu, F.; Liao, M.; Madhavan, V.; and Darrell, T. 2018. BDD100k: A diverse driving video database with scalable annotation tooling. In *arXiv:1805.04687*.

Yun, S.; Han, D.; Oh, S. J.; Chun, S.; Choe, J.; and Yoo, Y. 2019. CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features. In *ICCV*.

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

Zhu, X.; Pang, J.; Yang, C.; Shi, J.; and Lin, D. 2019. Adapting Object Detectors via Selective Cross-Domain Alignment. In *CVPR*.