Domain Adaptive Object Detection for Autonomous Driving under Foggy Weather

Jinlong Li[1], Runsheng Xu[2], Jin Ma[1], Qin Zou[3], Jiaqi Ma[2], Hongkai Yu[1*]
[1] Cleveland State University, [2] University of California, Los Angeles, [3] Wuhan University
j.li56@vikes.csuohio.edu, h.yu19@csuohio.edu

Abstract

Most object detection methods for autonomous driving usually assume a consistent feature distribution between training and testing data, which is not always the case when weathers differ significantly. The object detection model trained under clear weather might be not effective enough on the foggy weather because of the domain gap. This paper proposes a novel domain adaptive object detection framework for autonomous driving under foggy weather. Our method leverages both image-level and object-level adaptation to diminish the domain discrepancy in image style and object appearance. To further enhance the model's capabilities under challenging samples, we also come up with a new adversarial gradient reversal layer to perform adversarial mining for the hard examples together with domain adaptation. Moreover, we propose to generate an auxiliary domain by data augmentation to enforce a new domain-level metric regularization. Experimental results on public benchmarks show the effectiveness and accuracy of the proposed method. The code is available at `https://github.com/jinlong17/DA-Detect`.

1. Introduction

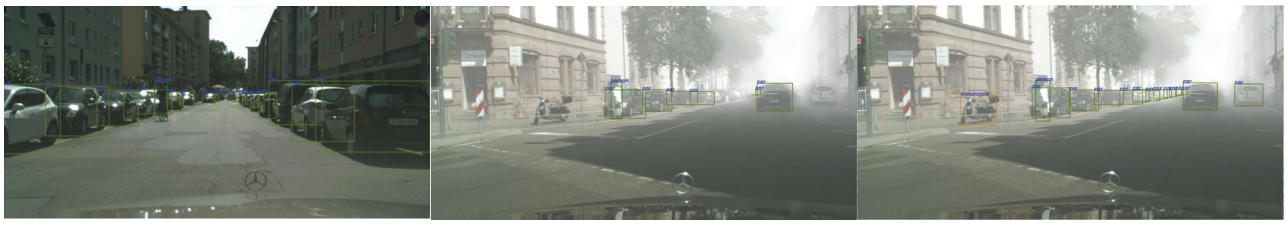
Autonomous driving has wide applications for intelligent transportation systems, such as improving the efficiency in the automatic 24/7 working manner, reducing the labor costs, enhancing the comfortableness of customers, and so on [23, 49]. With the computer vision and artificial intelligence techniques, object detection plays a critical role in autonomous driving to understand the surrounding driving scenarios [51, 56]. In some cases, the autonomous vehicle might work in the complex residential and industry areas. The diverse weather conditions might make the object detection in these environments more difficult. For example, the usages of heating, gas, coal, and vehicle emissions in residential and industry areas might be possible to generate more frequent foggy or hazy weather, leading to a significant challenge to the object detection system installed on the autonomous vehicle.

Many deep learning models such as Faster R-CNN [37], YOLO [36] have demonstrated great success in autonomous driving. However, most of these well-known methods assume that the feature distributions of training and testing data are homogeneous. Such an assumption may fail when taking the real-world diverse weather conditions into account [40]. For example, as shown in Fig. 1, the Faster R-CNN model trained on the clear-weather data (source domain) is capable of detecting objects accurately under good weather, but its performance drops significantly when it comes to the foggy weather (target domain). This degradation is caused by the feature domain gap between divergent weather conditions, as the model is unfamiliar with the feature distribution on the target domain, while the detection performance could be improved under the foggy weather with domain adaptation.

Domain adaptation, as a technique of transfer learning, is to reduce the domain shift between various weathers. This paper proposes a novel domain adaptation framework to achieve robust object detection performance in autonomous driving under foggy weather. As manually annotating images under adverse weathers is usually time-consuming, our design follows an unsupervised fashion same as that in [5, 26, 43], where clear-weather images (source domain) are well labeled and foggy weather images (target domain) have no annotations. Inspired by [5, 15], our method leverages both image-level and object-level adaptation to diminish the domain discrepancy in image style and object appearance jointly, which is realized by involving image-level and object-level domain classifiers to enable our convolutional neural networks generating domain-invariant latent feature representations. Specifically, the domain classifiers aim to maximize the probability of distinguishing the features produced by different domains, whereas the detection model expects to generate the domain-invariant features to confuse the classifiers.

This paper also addresses two critical insights that are ignored by previous domain adaptation methods [5, 9, 15, 26, 58]: 1) Different training samples might have different challenging levels to be fully harnessed during the transfer learning, while existing works usually omit such diversity;

*Corresponding author



(a) (b) (c)

Figure 1: Illustration of the domain adaptive object detection for autonomous driving: (a) Faster R-CNN [37] detection under clear weather, (b) Faster R-CNN detection under foggy weather without domain adaptation, (c) Faster R-CNN detection under foggy weather with the proposed domain adaptation.

2) Previous domain adaptation methods only consider the source domain and target domain for transfer learning, while the domain-level feature metric distance to the third related domain might be neglected. However, embedding the mining for hard examples and involving an extra related domain might potentially further enhance the model's robust learning capabilities, which has not been carefully explored before. To emphasize these two insights, we propose a new Adversarial Gradient Reversal Layer (AdvGRL) and generate an auxiliary domain by data augmentation. The AdvGRL performs adversarial mining for the hard examples to enhance the model learning on the challenging scenarios, and the auxiliary domain enforces a new domain-level metric regularization during the transfer learning. Experimental results on the public benchmarks Cityscapes [7] and Foggy Cityscapes [40] show the effectiveness of each proposed component and the superior object detection performance over the baseline and comparison methods. Overall, the contributions of this paper are summarized as follows:

- We propose a novel deep transfer learning based domain adaptive object detection framework for autonomous driving under foggy weather, including the image-level and object-level adaptations, which is trained with labeled clear-weather data and unlabeled foggy-weather data to enhance the generalization ability of the deep learning based object detection model.

- We propose a new Adversarial Gradient Reversal Layer (AdvGRL) to perform adversarial mining for the hard examples together with the domain adaptation to further enhance the model's transfer learning capabilities under challenging samples.

- We propose a new domain-level metric regularization during the transfer learning. By generating an auxiliary domain with data augmentation, the domain-level metric constraint between source domain, auxiliary domain, and target domain is ensured as regularization during the transfer learning.

2. Related Work

2.1. Object detection for autonomous driving

Recent advancement in deep learning has brought outstanding progress in autonomous driving [6, 25, 33, 50], and object detection has been one of the most active topic under this field [8, 41, 56]. Regarding the network architecture, current object detection algorithms can be roughly split into two categories: two-stage methods and single-stage methods. Two-stage object detection algorithms typically compose of two processes: 1) region proposal, 2) object classification and localization refinement. R-CNN [14] is the first work for this kind of methods, it applies selective search for regional proposals and independent CNNs for each object prediction. Fast R-CNN [13] improves R-CNN by obtaining object features from the shared feature map learned by one CNN. Faster R-CNN [37] further enhances the framework by proposing Region Proposal Network (RPN) to replace the selective search stage. Single-stage object detection algorithms predict object bounding boxes and classes simultaneously in one same stage. These methods usually leverage pre-defined anchors to classify objects and regress bounding boxes, they are less time-consuming but less accurate compared to two-stage algorithms. Milestones for this category include SSD-series [29], YOLO-series [36] and RetinaNet [28]. Despite their success in clear-weather visual scenes, these object detection methods might not be employed in autonomous driving directly due to the complex real-world weather conditions.

2.2. Object detection for autonomous driving under different weather

In order to address the diverse weather conditions encountered in autonomous driving, many datasets have been generated [31, 32, 34, 40] and many methods have been proposed [2, 17, 18, 22, 35, 42] in recent years. For example, Foggy Cityscape [40] is a synthetic dataset that applies fog simulation to Cityscape for scene understanding in foggy weather. TJU-DHD [32] is a diverse dataset for object detection in real-world scenarios which contains variances in terms of illumination, scene, weather and season. In this

paper, we focus on the object detection problem in foggy weather. Huang et al. [22] propose a DSNet (Dual-Subnet Network) that involves a detection subnet and a restoration subnet. This network can be trained with multi-task learning by combining visibility enhancement task and object detection task, thus outperforms pure object detectors. Hahner et al. [17] develop a fog simulation approach to enhance existing real lidar dataset, and show this approach can be leveraged to improve current object detection methods in foggy weather. Qian et al. [35] propose a MVDNet (Multi-modal Vehicle Detection Network) that takes advantage of lidar and radar signals to obtain proposals. Then the region-wise features from these two sensors are fused together to get final detection results. Bijelic et al. [2] develop a network that takes the data from four sensors as input: lidar, RGB camera, gated camera, and radar. This architecture uses entropy-steered adaptive deep fusion to get fused feature maps for prediction. These methods typically rely on input data from other sensors rather than RGB camera itself, which is not the general case for many autonomous driving cars. Thus we aim to develop an object detection architecture that only takes RGB camera data as input in this work.

2.3. Domain adaptation for object detection

Domain adaptation reduces the discrepancy between different domains, thus allows the model trained on source domain to be applicable on unlabeled target domain. Previous domain adaptation works mainly focus on the task of image classification [44–46, 53], while more and more methods have been proposed to solve domain adaptation for object detection in recent years [5,15,24,39,47,48,52,55,57]. Domain adaptive detectors could be obtained if the features from different domains are aligned [5, 15, 18, 39, 47]. From this perspective, Chen et al. [5] introduce a Domain Adaptive Faster R-CNN framework to reduce domain gap from image level and instance level, and the image-and-instance consistency is subsequently employed to improve cross-domain robustness. He et al. [18] propose a MAF (multi-adversarial Faster R-CNN) framework to minimize the domain distribution disparity by aligning domain features and proposal features hierarchically. On the other hand, some works try to solve domain adaptation through image style transfer methods [21, 24, 41]. Shan et al. [41] first convert images from source domain to target domain with image translation module, then train the object detector with adversarial training on target domain. Hsu et al. [21] choose to translate images progressively, and add a weighted task loss during adversarial training stage for tackling the problem of image quality difference. Many previous methods [4, 27, 38, 59] design complex architectures. [59] used multi-scale backbone Feature Pyramid Networks and considered pixel-level and category-level adaptation. [27] used the complex Graph Convolution Network and graph matching algorithms. [38] used the similarity-based clustering and grouping. [4] uses the uncertainty-guided self-training mechanism (Probabilistic Teacher and Focal Loss) to capture the uncertainty of unlabeled target data from a gradually evolving teacher and guides student learning. Differently, our method does not bring extra learnable parameters to original Faster R-CNN model because our AdvGRL is based on adversarial training (gradient reversal) and Domain-level Metric Regularization is based on triplet loss. Previous domain adaptation methods usually treat training samples at the same challenging level, while we employ advGRL for adversarial hard example mining to improve transfer learning. Moreover, we generate an auxiliary domain and apply domain-level metric regularization to avoid overfitting.

3. Proposed Method

In this section, we will first introduce the overall network architecture, then describe the image-level and object-level adaptation method, and finally, reveal the details of AdvGRL and domain-level metric regularization.

3.1. Network Architecture

As illustrated in Fig. 2, our proposed model adopts the pipeline in Faster R-CNN for object detection. The Convolutional Neural Network (CNN) backbone extracts the image-level features from the RGB images and send them to Region Proposal Network (RPN) to generate object proposals. Afterwards, the ROI pooling accepts both image-level features and object proposals as the input to retrieve the object-level features. Eventually, a detection head is applied on the object-level features to produce the final predictions. Based on the Faster R-CNN framework, we integrate two more components: image-level domain adaptation module, and object-level domain adaptation module. For both modules, we deploy a new Adversarial Gradient Reversal Layer (AdvGRL) together with the domain classifier to extract domain-invariant features and perform adversarial hard example mining. Moreover, we involve an auxiliary domain to impose a new domain-level metric regularization to enforce the feature metric distance between different domains. All three domains, *i.e.*, source, target, and auxiliary domains, will be employed simultaneously during the training.

3.2. Image-level Adaptation

The image-level domain representation is obtained from the backbone feature extraction and contains rich global information such as style, scale and illumination, which can potentially pose significant impacts on the detection task [5]. Therefore, a domain classifier is introduced to classify the domains of the upcoming image-level features to enhance the image-level global alignment. The domain classifier is just a simple CNN with two convolutional layers and it will output a prediction to identify the feature domain. We use

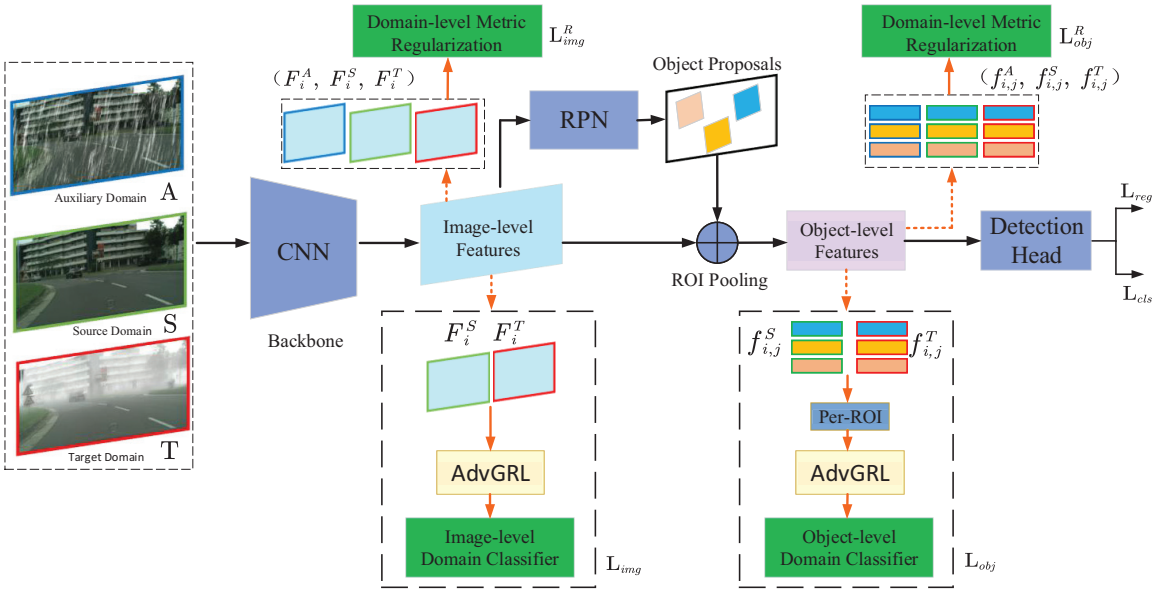


Figure 2: The architecture of proposed domain adaptive object detection for autonomous driving under foggy weather. Based on the traditional Faster R-CNN architecture [37], the image-level and object-level domain adaptations with adversarial gradient reversal layer (AdvGRL) and domain-level metric regularization are designed in the proposed framework. This figure is best viewed in color.

the binary cross entropy loss for the domain classifier as follows:

$$L_{img} = -\sum_{i=1}^{N}[G_i \log P_i + (1-G_i)\log(1-P_i)], \quad (1)$$

where $i \in \{1, ..., N\}$ represents the N training images, $G_i \in \{1, 0\}$ is the ground truth of the domain label in the i-th training image (1 and 0 stand for source and target domains respectively), and P_i is the prediction of the domain classifier.

3.3. Object-level Adaptation

Besides the image-level global difference in different domains, the objects in different domains might be also dissimilar in the appearance, size, color, *etc*. In this paper, we define each region proposal after the ROI Pooling layer in Faster R-CNN as a potential object. Similar with image-level adaptation module, after retrieving the object-level domain representation by ROI pooling, we implement a object-level domain classifier to identify the feature derivation from local information. A well-trained object-level classifier, a neural network with 3 fully-connected layers, will help align the object-level feature distribution. We also use the binary cross entropy loss for this domain classifier:

$$L_{obj} = -\sum_{i=1}^{N}\sum_{j=1}^{M}[G_{i,j} \log P_{i,j} + (1-G_{i,j})\log(1-P_{i,j})], \quad (2)$$

where $j \in \{1, ..., M\}$ is the j-th detected object (region proposal) in the i-th image, $P_{i,j}$ is the prediction of the object-level domain classifier for the j-th region proposal in the i-th image, and $G_{i,j}$ is the corresponding binary ground-truth label for source and target domains respectively.

3.4. Adversarial Gradient Reversal Layer

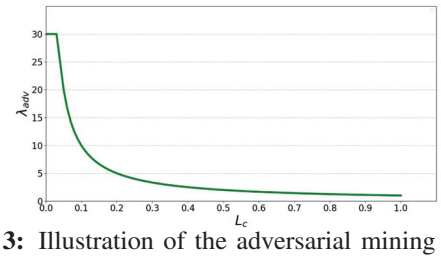


Figure 3: Illustration of the adversarial mining for hard training examples by the proposed AdvGRL. In this example, we set $\lambda_0 = 1$, $\beta = 30$. Harder training examples with lower domain classifier loss L_c will have larger response.

In this section, we first review the original Gradient Reversal Layer (GRL) [10], then we make a detailed description of the proposed Adversarial Gradient Reversal Layer (AdvGRL) for our domain adaptive object detection framework.

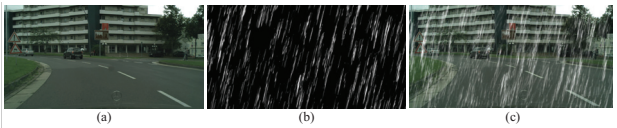


Figure 4: The example of generating auxiliary domain using Cityscapes dataset: (a) an original image, (b) a rain map by RainMix [16], (c) a synthetic rainy Cityscapes image for the auxiliary domain.

The original GRL is used for unsupervised domain adaptation of the image classification task [10]. Specifically, it leaves the input unchanged during forward propagation and reverses the gradient by multiplying it by a negative scalar when back-propagating to the base network ahead during training. A domain classifier is trained to maximize the probability of identifying the domain while the base network ahead is optimized to confuse the domain classifier. In this way, the domain-invariant features are obtained to realize the domain adaptation. The forward propagation of GRL is defined as:

$$R_\lambda(\mathbf{v}) = \mathbf{v}, \tag{3}$$

where $\mathbf{v}$ is an input feature vector, and R_λ denotes the forwarding function that GRL performs, and the back-propagation of GRL is defined as:

$$\frac{dR_\lambda}{d\mathbf{v}} = -\lambda\mathbf{I}, \tag{4}$$

where $\mathbf{I}$ is an identity matrix and $-\lambda$ is a negative scalar.

The original GRL sets either a constant or a changing $-\lambda$ based on the training iterations [10]. However, this setting ignores an insight that different training samples might have different challenging levels during the transfer learning. Therefore, this paper proposes a novel AdvGRL to perform adversarial mining for the hard examples together with the domain adaptation to further enhance the model's transfer learning capabilities under challenging examples. This can be done by simply replacing λ by a new λ_{adv} in Eq. (4) of GRL, which forms the proposed AdvGRL. Particularly, λ_{adv} is calculated as:

$$\lambda_{adv} = \begin{cases} \min(\frac{\lambda_0}{L_c}, \beta), & L_c < \alpha \\ \lambda_0, & \text{otherwise}, \end{cases} \tag{5}$$

where L_c is the loss of the domain classifier, α is a hardness threshold to judge whether the training sample is challenging, β is the overflow threshold to avoid generating excessive gradients in the back-propagation, and $\lambda_0 = 1$ is set as a fixed parameter in our experiment. In other words, if the domain classifier's loss L_c is smaller, the domain of the training sample can be more easily identified, whose feature is not the desired domain-invariant feature, so this kind of training sample is a harder example for domain adaptation. The relation of λ_{adv} and L_c is shown in Fig. 3.

On summary, the proposed AdvGRL has two effects: 1) AdvGRL could use negative gradients during back-propagation to confuse the domain classifier so as to generate domain-invariant features; 2) AdvGRL could perform adversarial mining for the hard examples to further enhance the model generalization under challenging examples. The proposed AdvGRL is applied to both image-level and object-level domain adaptation in our domain adaptive object detection framework, as shown in Fig. 2.

3.5. Domain-level Metric Regularization

Previous existing domain adaptation methods mainly focus on the transfer learning from source domain S to target domain T, which neglects the potential benefits of the third related domain can bring. To address this and thus additionally involve the feature metric constraint between different domains, we introduce an auxiliary domain for a domain-level metric regularization during the transfer learning.

Based on the source domain S, we can apply some advanced data augmentation methods to generate an auxiliary domain A. For the autonomous driving scenario, the training data in different weather conditions can be synthesized from the clear-weather data, then the three input images of our architecture (as shown in Fig. 2) could be aligned images. For example, we generate an auxiliary domain with the advanced data augmentation method RainMix [16, 20]. Specifically, we randomly sample a rain map from the public dataset of real rain streaks [11], then perform random transformations using the RainMix technique on the rain map, where these random transformations (*i.e.*, rotate, zoom, translate, shear) are sampled and combined. Finally, these transformed rain maps can be blended with the source domain images, which can simulate the diverse rain patterns in the real world. The example of generating auxiliary domain is shown in Fig. 4. Different with other methods including data augmentation to the source/target domain, by generating an auxiliary domain with data augmentation, the domain-level metric constraint between source, auxiliary, and target domains is ensured.

Let us define the i-th training image's global image-level features of S, A and T as F_i^S, F_i^A, and F_i^T respectively. We expect to ensure the feature metric distance between the F_i^S and F_i^T closer than the feature metric distance between F_i^S and F_i^A after reducing the domain gap between S and T, which is defined as:

$$d(F_i^S, F_i^T) < d(F_i^S, F_i^A), \tag{6}$$

where $d(,)$ denotes the metric distance of the corresponding features. This constraint can be implemented by a triplet structure, where the F_i^S, F_i^T, F_i^A can be treated as anchor, positive and negative in the triplet structure. Therefore, as the domain-level metric regularization on image features,

the above image-level constraint in Eq. (6) is equivalent to minimize the following image-level triplet loss:

$$L_{img}^{R} = \max(d(F_i^S, F_i^T) - d(F_i^S, F_i^A) + \delta, 0), \quad (7)$$

where the parameter δ is used as a margin constraint and we set $\delta = 1.0$ in our experiments.

Similarly, let us define the i-th training image's j-th object-level features of S, A and T as $f_{i,j}^S$, $f_{i,j}^A$, and $f_{i,j}^T$ respectively. As the domain-level metric regularization on object features, we will also minimize the following object-level triplet loss:

$$L_{obj}^{R} = \max(d(f_{i,j}^S, f_{i,j}^T) - d(f_{i,j}^S, f_{i,j}^A) + \delta, 0). \quad (8)$$

3.6. Loss Function

The final training loss of the proposed network is a summation of each individual part, which can be written as:

$$L = L_{cls} + L_{reg} + w * (L_{img} + L_{obj} + L_{img}^{R} + L_{obj}^{R}), \quad (9)$$

where L_{cls} and L_{reg} are the loss of classification and the loss of regression in the original Faster R-CNN respectively, and w is a weight to balance the Faster R-CNN loss and the domain adaptation loss for training. We set $w = 0.1$ in our experiments. In the training, the proposed domain adaptive object detection framework can be trained in an end-to-end manner using a standard Stochastic Gradient Descent algorithm. During the testing, the original Faster R-CNN architecture with trained adapted weights can be used for object detection, after removing the domain adaptation components.

3.7. General Domain Adaptive Object Detection

Our model has the capability to be adapted to general domain adaptive object detection. For the scenarios that the images from target domain are synthesized from the source domain with pixel-to-pixel correspondence (*e.g.*, Cityscapes⟶Foggy Cityscapes), our method can be directly applied without modification. For the scenarios where target and source domains do not have strict correspondence (*e.g.*, Cityscapes⟶KITTI), our method can be applied by simply removing the L_{obj}^{R} loss to eliminate the dependence on the object alignment in the model training.

4. Experiments

4.1. Benchmark

Our experiments are based on the public object detection benchmarks Cityscapes [7] and Foggy Cityscapes [40] for autonomous driving. Cityscapes [7] is a widely used autonomous driving dataset, which is a collection of images with city street scenarios in clear weather conditions from 27 cities. In Cityscapes dataset, there are 2,975 training images and 500 validation images with instance segmentation annotations which can be transformed into bounding-box annotations with 8 categories. All images are 3-channel RGB images and captured by a car-mounted video camera with the same resolution of 1024×2048. Foggy Cityscapes [40] is established by simulating the fog of different intensity levels on the Cityscapes images, which generates the simulated three levels of fog based on the depth map and a physical model [40]. Its image number, resolution, training/validation split, and annotations are same as those of Cityscapes. Following the previous methods [5, 15, 47], the images with the fog of highest intensity level are utilized as the target domain for transfer learning in our experiments.

4.2. Experimental Setting

Dataset setting: We set the labeled training set of Cityscapes [7] as source domain and the unlabeled training set of Foggy Cityscapes [40] as target domain during the training. Then, the trained model is tested on the validation set of Foggy Cityscapes to report the evaluation result. We denote this setting as the Cityscapes⟶Foggy Cityscapes experiment in this paper.

Training and parameter setting: In the experiments, we adopt ResNet-50 as the backbone for the Faster R-CNN [37] detection network, which is pre-trained on ImageNet. During training, following setting in [5, 37], the back-propagation and stochastic gradient descent (SGD) are used to optimize all the networks. The whole network is trained with an initial learning rate 0.01 for $50k$ iterations and then reduced to 0.001 for another $20k$ iterations. For all experiments, a weight decay of 0.0005 and a momentum of 0.9 are used, and each batch includes an image of source domain, an image of target domain and an image of auxiliary domain. For comparison, the λ in the original GRL (Eq. (4)) is set as 1. The hardness threshold α in the AdvGRL (Eq. (5)) is set as 0.63 by averaging the values of Eq. (1) when $P_i = 0.7, G_i = 1$ and $P_i = 0.3, G_i = 0$. Our code is implemented with PyTorch and Mask R-CNN Benchmark Toolbox [30], and all models are trained using a GeForce RTX3090 GPU card with 24GB memory.

Evaluation metrics and comparison methods: We set the Intersection over Union (IoU) threshold as 0.5 to compute the Average Precision (AP) of each category and mean Average Precision (mAP) of all categories. Then we compare our proposed method with some recent domain adaptation comparison methods in our experiments, such as SCDA [61], DM [24], MAF [18], MCAR [57], SWDA [39], PDA [21], RPN-PR [55], MTOR [3], DA-Faster [5], GPA [47], and UaDAN [15].

Table 1: AP for each class and overall mAP with comparison methods on the Cityscapes⟶Foggy Cityscapes experiment (%) as clear to foggy adaptation. Note that the best performance is bold and the second best is underlined.

Methods	bus	bicycle	car	mcycle	person	rider	train	truck	mAP
SCDA-CVPR'19 [61]	39.0	33.6	48.5	28.0	33.5	38.0	23.3	26.5	33.8
DM-CVPR'19 [24]	38.4	32.2	44.3	28.4	30.8	40.5	34.5	27.2	34.6
MAF-ICCV'19 [18]	39.9	33.9	43.9	29.2	28.2	39.5	33.3	23.8	34.0
MCAR-ECCV'20 [57]	44.1	36.6	43.9	**37.4**	32.0	42.1	43.4	**31.3**	38.8
SWDA-CVPR'19 [39]	36.2	35.3	43.5	30.0	29.9	42.3	32.6	24.5	34.3
PDA-WACV'20 [21]	44.1	35.9	**54.4**	29.1	36.0	45.5	25.8	24.3	36.9
MTOR-CVPR-19 [3]	38.6	35.6	44.0	28.3	30.6	41.4	40.6	21.9	35.1
DA-Faster-CVPR'18 [5]	49.8	39.0	53.0	28.9	35.7	45.2	45.4	30.9	41.0
GPA-CVPR'20 [47]	45.7	38.7	54.1	32.4	32.9	**46.7**	41.1	24.7	39.5
RPN-PR-CVPR'21 [55]	43.6	36.8	50.5	29.7	33.3	45.6	42.0	30.4	39.0
UaDAN-TMM'21 [15]	49.4	38.9	53.6	32.3	**36.5**	46.1	42.7	28.9	41.1
Ours w/o Auxiliary Domain	48.4	36.7	53.5	26.1	36.1	45.9	39.1	29.3	40.2
Ours	**51.2**	**39.1**	54.3	31.6	**36.5**	**46.7**	**48.7**	30.3	**42.3**
Oracle	49.9	45.8	65.2	39.6	46.5	51.3	34.2	32.6	45.6

4.3. Clear to Foggy Adaptation

The results of weather adaptation from clear weather to foggy weather are represented in Table 1. Compared with other domain adaptation methods, we can see that our proposed method achieves the best detection performance with a mAP of 42.3%, which is higher than the second best method UaDAN [15] with a mAP improvement of 1.2%. For each category, we can see that the proposed method is able to alleviate the domain gap over most of the categories in Foggy Cityscapes, *e.g.*, bus got 51.2%, bicycle got 39.1%, person got 36.5%, rider got 46.7%, and train got 48.7% as the best performance in AP, which is highlight in Table 1. The proposed method can reach the 48.7% AP for the train detection in Foggy Cityscapes, compared to the 45.4% AP by the second best method DA-Faster, where the proposed method is 3.3% better than DA-Faster. While PDA got 54.4% in car, GPA got 32.4% in motorcycle, DA-faster got 30.9% in truck as the best performance in some categories, the proposed method is comparable across these three categories with a minor difference. Obviously, compared to these recent domain adaptation methods, the proposed method achieves the best performance in overall mAP performance and more than half categories of Foggy Cityscapes.

Table 2: Ablation study for mAP on the Cityscapes⟶Foggy Cityscapes experiment.

	img	obj	AdvGRL	Reg	mAP
Source only					23.41
img+GRL	✓				38.10
obj+GRL		✓			38.02
img+obj+GRL (Baseline)	✓	✓			38.43
img+obj+AdvGRL	✓	✓	✓		40.23
img+obj+GRL+Reg	✓	✓		✓	41.97
img+obj+AdvGRL+Reg	✓	✓	✓	✓	**42.34**

4.4. Cross-Camera Adaptation

To fully evaluate the proposed method, we conduct an experiment to perform the cross-camera adaptation between real-world autonomous driving datasets with different camera settings. To apply our method to the unaligned datasets in the real-world, we simply remove L_{obj}^{R} (Eq. 8) to apply our method from Cityscapes (source) to KITTI [12] (target) datasets for cross-camera adaptation. Same as [5], we use KITTI training set (7,481 images of resolution 1250×375) as target domain in both adaptation and evaluation, and AP of Car on target domain is evaluated. The result is in Table 3, where the proposed method achieved outstanding performance compared with recent SOTA methods.

4.5. Ablation Study on Components

The effect of each individual proposed component for the domain adaptation detection method is investigated in this section. All experiments are conducted with the same RestNet-50 backbone on the Cityscapes⟶Foggy Cityscapes experiment. The results are presented in Table 2. In the first row, 'img' and 'obj' stand for the image-level adaptation module and object-level adaptation module respectively, while 'AdvGRL' and 'Reg' denote the proposed Adversarial Gradient Reversal Layer and domain-level metric Regularization respectively. 'img+obj+GRL' stands for the baseline model in our experiment. We denote that 'img+obj+AdvGRL' (Ours w/o Auxiliary Domain) and 'img+obj +AdvGRL+Reg' use the AdvGRL to replace the original GRL. The 'Source only' indicates the Faster R-CNN model without domain adaptation trained only with labeled source domain images. The ablation study in Table 2 clearly justifies the positive effect of each proposed component of the domain adaptive object detection.

Table 3: AP of Car on the Cityscapes⟶KITTI experiment as cross-camera adaptation.

	MAF-ICCV'19 [18]	ATF-ECCV'20 [19]	ART-CVPR'20 [58]	GPA-CVPR'20 [47]	SGA-TMM'21 [54]	UIT-ESwA'22 [1]	Ours
AP	72.10	73.50	73.60	65.36	72.02	73.70	**74.38**

Figure 5: Hard examples (bigger λ_{adv}) mined by AdvGRL. Left to right: two mined hard examples, an easy example.

4.6. Ablation Study on Parameters

The study on different hyper-parameters of Eq. 9 and Eq. 5 are conducted. We use the Cityscapes⟶Foggy Cityscapes as the study case. First, the loss balance weight w in Eq. 9 is set as 0.1, 0.01, 0.001 separately for training, and the corresponding detection mAP(s) are 42.34, 41.30, 41.19, respectively. Second, in the AdvGRL (Eq. 5), the overflow threshold β and hardness threshold α are set as (1) $\beta = 30, \alpha = 0.63$, (2) $\beta = 10, \alpha = 0.63$, (3) $\beta = 30, \alpha = 0.54$, and (4) $\beta = 10, \alpha = 0.54$, where $\alpha = 0.54$ is computed by averaging the values of Eq. 1 when $P_i = 0.9, G_i = 1$ and $P_i = 0.1, G_i = 0$. The detection mAP(s) of these settings are (1) 42.34, (2) 38.83, (3) 39.38, (4) 40.47, respectively.

4.7. Discussion on Visualized Hard Examples

Using λ_{adv} of the proposed AdvGRL, we could find some hard examples, as shown in Fig. 5. We compute the L_1 distance of features F_i^S and F_i^T after the CNN backbone of Fig. 2 as the approximated hardness ah, where smaller ah means harder for transfer learning. Intuitively, if the fog covers more objects as shown in bounding-box regions of Fig. 5, it will be more difficult.

4.8. Discussion on Domain Randomization, Pre-trained Models, and Qualitative Results

Domain Randomization: Domain randomization might be used to reduce the domain shift between source domain and target domain. We use two ways as the domain randomization in the Cityscapes⟶Foggy Cityscapes experiment, *i.e.*, regular data argumentation and CycleGAN [60] based image style transfer. 1) We construct the auxiliary domain by regular data argumentation (color change + blur + salt & pepper noises), where our method's detection mAP is 38.7, compared to our 42.3 by the auxiliary domain with rain synthesis. 2) We train a CycleGAN to transfer the image style between the training sets of Cityscapes and Foggy Cityscapes. Using the generated fake foggy-style image of Cityscapes by the trained CycleGAN model to train a Faster R-CNN model, it achieves detection mAP as 32.8. These experiments show that commonly used domain randomization could not well solve the domain adaptation problem.

Pre-trained Models: We use the pre-trained Faster R-CNN model in [5] to initialize our method, then our method gets the detection mAP as 41.3 in the Cityscapes⟶Foggy Cityscapes experiment, compared to 42.3 by our method without pre-trained detection model.

Qualitative Results: We visualize some detection results on the Foggy Cityscapes dataset in Fig. 6, which shows that the proposed domain adaptive method improves the detection performance in foggy weather significantly.

(a) (b) (c)

Figure 6: Qualitative visualization results on validation set of Foggy Cityscapes: (a) Original Faster R-CNN without domain adaptation, (b) Faster R-CNN with image-level and object-level adaptations using GRL (Baseline), (c) Proposed Method. Note: different colors represent different categories.

5. Conclusions

In this paper, we propose a novel domain adaptive object detection framework for autonomous driving. The image-level and object-level adaptations are designed to reduce the domain shift on the global image style and local object appearance. A new adversarial gradient reversal layer is proposed to perform adversarial mining for hard examples together with domain adaptation. Considering the feature metric distance between source domain, target domain and auxiliary domain by data augmentation, we propose a new domain-level metric regularization. Furthermore, our method could be applied to solve the general domain adaptive object detection problem. We conduct the transfer learning experiments from Cityscapes to Foggy Cityscapes and from Cityscapes to KITTI, and experimental results show that the proposed method is quite effective.

Acknowledgement: This work was supported by NSF 2215388.

References

[1] Vinicius F Arruda, Rodrigo F Berriel, Thiago M Paixão, Claudine Badue, Alberto F De Souza, Nicu Sebe, and Thiago Oliveira-Santos. Cross-domain object detection using unsupervised image translation. *Expert Systems with Applications*, 192:116334, 2022.

[2] Mario Bijelic, Tobias Gruber, Fahim Mannan, Florian Kraus, Werner Ritter, Klaus Dietmayer, and Felix Heide. Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 11682–11692, 2020.

[3] Qi Cai, Yingwei Pan, Chong-Wah Ngo, Xinmei Tian, Lingyu Duan, and Ting Yao. Exploring object relation in mean teacher for cross-domain detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 11457–11466, 2019.

[4] Meilin Chen, Weijie Chen, Shicai Yang, Jie Song, Xinchao Wang, Lei Zhang, Yunfeng Yan, Donglian Qi, Yueting Zhuang, Di Xie, et al. Learning domain adaptive object detection with probabilistic teacher. In *International Conference on Machine Learning*, pages 3040–3055. PMLR, 2022.

[5] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3339–3348, 2018.

[6] Henrik Christensen, David Paz, Hengyuan Zhang, Dominique Meyer, Hao Xiang, Yunhai Han, Yuhan Liu, Andrew Liang, Zheng Zhong, and Shiqi Tang. Autonomous vehicles for micro-mobility. *Autonomous Intelligent Systems*, 1(1):1–35, 2021.

[7] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3213–3223, 2016.

[8] Di Feng, Ali Harakeh, Steven L Waslander, and Klaus Dietmayer. A review and comparative study on probabilistic object detection in autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[9] Lan Fu, Hongkai Yu, Felix Juefei-Xu, Jinlong Li, Qing Guo, and Song Wang. Let there be light: Improved traffic surveillance via detail preserving night-to-day transfer. *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[10] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning*, pages 1180–1189. PMLR, 2015.

[11] Kshitiz Garg and Shree K Nayar. Photorealistic rendering of rain streaks. *ACM Transactions on Graphics*, 25(3):996–1002, 2006.

[12] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3354–3361. IEEE, 2012.

[13] Ross Girshick. Fast r-cnn. In *IEEE International Conference on Computer Vision*, pages 1440–1448, 2015.

[14] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014.

[15] Dayan Guan, Jiaxing Huang, Aoran Xiao, Shijian Lu, and Yanpeng Cao. Uncertainty-aware unsupervised domain adaptation in object detection. *IEEE Transactions on Multimedia*, 2021.

[16] Qing Guo, Jingyang Sun, Felix Juefei-Xu, Lei Ma, Xiaofei Xie, Wei Feng, Yang Liu, and Jianjun Zhao. Efficientderain: Learning pixel-wise dilation filtering for high-efficiency single-image deraining. In *AAAI Conference on Artificial Intelligence*, pages 1487–1495, 2021.

[17] Martin Hahner, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Fog simulation on real lidar point clouds for 3d object detection in adverse weather. In *IEEE International Conference on Computer Vision*, pages 15283–15292, 2021.

[18] Zhenwei He and Lei Zhang. Multi-adversarial faster-rcnn for unrestricted object detection. In *IEEE International Conference on Computer Vision*, pages 6668–6677, 2019.

[19] Zhenwei He and Lei Zhang. Domain adaptive object detection via asymmetric tri-way faster-rcnn. In *European conference on computer vision*, pages 309–324. Springer, 2020.

[20] Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. AugMix: A simple data processing method to improve robustness and uncertainty. *International Conference on Learning Representations*, 2020.

[21] Han-Kai Hsu, Chun-Han Yao, Yi-Hsuan Tsai, Wei-Chih Hung, Hung-Yu Tseng, Maneesh Singh, and Ming-Hsuan Yang. Progressive domain adaptation for object detection. In *IEEE Winter Conference on Applications of Computer Vision*, pages 749–757, 2020.

[22] Shih-Chia Huang, Trung-Hieu Le, and Da-Wei Jaw. Dsnet: Joint semantic learning for object detection in inclement weather conditions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(8):2623–2633, 2020.

[23] Kichun Jo, Junsoo Kim, Dongchul Kim, Chulhoon Jang, and Myoungho Sunwoo. Development of autonomous car—part i: Distributed system architecture and development process. *IEEE Transactions on Industrial Electronics*, 61(12):7131–7140, 2014.

[24] Taekyung Kim, Minki Jeong, Seunghyeon Kim, Seokeon Choi, and Changick Kim. Diversify and match: A domain adaptive representation learning paradigm for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 12456–12465, 2019.

[25] E Li, Shuaijun Wang, Chengyang Li, Dachuan Li, Xiangbin Wu, and Qi Hao. Sustech points: A portable 3d point cloud interactive annotation platform system. In *IEEE Intelligent Vehicles Symposium*, pages 1108–1115, 2020.

[26] Jinlong Li, Zhigang Xu, Lan Fu, Xuesong Zhou, and Hongkai Yu. Domain adaptation from daytime to nighttime: A situation-sensitive vehicle detection and traffic flow parameter estimation framework. *Transportation Research Part C: Emerging Technologies*, 2021.

[27] Wuyang Li, Xinyu Liu, and Yixuan Yuan. Sigma: Semantic-complete graph matching for domain adaptive object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5291–5300, 2022.

[28] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision*, pages 2980–2988, 2017.

[29] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*, pages 21–37. Springer, 2016.

[30] Francisco Massa and Ross Girshick. maskrcnn-benchmark: Fast, modular reference implementation of Instance Segmentation and Object Detection algorithms in PyTorch. `https://github.com/facebookresearch/maskrcnn-benchmark`, 2018.

[31] Claudio Michaelis, Benjamin Mitzkus, Robert Geirhos, Evgenia Rusak, Oliver Bringmann, Alexander S Ecker, Matthias Bethge, and Wieland Brendel. Benchmarking robustness in object detection: Autonomous driving when winter is coming. *arXiv preprint arXiv:1907.07484*, 2019.

[32] Yanwei Pang, Jiale Cao, Yazhao Li, Jin Xie, Hanqing Sun, and Jinfeng Gong. Tju-dhd: A diverse high-resolution dataset for object detection. *IEEE Transactions on Image Processing*, 30:207–219, 2020.

[33] David Paz, Hengyuan Zhang, Qinru Li, Hao Xiang, and Henrik I Christensen. Probabilistic semantic mapping for urban autonomous driving applications. In *IEEE International Conference on Intelligent Robots and Systems*, pages 2059–2064. IEEE, 2020.

[34] Quang-Hieu Pham, Pierre Sevestre, Ramanpreet Singh Pahwa, Huijing Zhan, Chun Ho Pang, Yuda Chen, Armin Mustafa, Vijay Chandrasekhar, and Jie Lin. A* 3d dataset: Towards autonomous driving in challenging environments. In *IEEE International Conference on Robotics and Automation*, pages 2267–2273. IEEE, 2020.

[35] Kun Qian, Shilin Zhu, Xinyu Zhang, and Li Erran Li. Robust multimodal vehicle detection in foggy weather using complementary lidar and radar signals. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 444–453, 2021.

[36] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016.

[37] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 28, 2015.

[38] Farzaneh Rezaeianaran, Rakshith Shetty, Rahaf Aljundi, Daniel Olmeda Reino, Shanshan Zhang, and Bernt Schiele. Seeking similarities over differences: Similarity-based domain alignment for adaptive object detection. In *IEEE/CVF International Conference on Computer Vision*, pages 9204–9213, 2021.

[39] Kuniaki Saito, Yoshitaka Ushiku, Tatsuya Harada, and Kate Saenko. Strong-weak distribution alignment for adaptive object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6956–6965, 2019.

[40] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Semantic foggy scene understanding with synthetic data. *International Journal of Computer Vision*, 126(9):973–992, 2018.

[41] Yuhu Shan, Wen Feng Lu, and Chee Meng Chew. Pixel and feature level based domain adaptation for object detection in autonomous driving. *Neurocomputing*, 367:31–38, 2019.

[42] Vishwanath A Sindagi, Poojan Oza, Rajeev Yasarla, and Vishal M Patel. Prior-based domain adaptive object detection for hazy and rainy conditions. In *European Conference on Computer Vision*, pages 763–780. Springer, 2020.

[43] Shaoyue Song, Hongkai Yu, Zhenjiang Miao, Jianwu Fang, Kang Zheng, Cong Ma, and Song Wang. Multi-spectral salient object detection by adversarial domain adaptation. In *AAAI Conference on Artificial Intelligence*, pages 12023–12030, 2020.

[44] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 7167–7176, 2017.

[45] Mei Wang and Weihong Deng. Deep visual domain adaptation: A survey. *Neurocomputing*, 312:135–153, 2018.

[46] Ni Xiao and Lei Zhang. Dynamic weighted learning for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 15242–15251, 2021.

[47] Minghao Xu, Hang Wang, Bingbing Ni, Qi Tian, and Wenjun Zhang. Cross-domain detection via graph-induced prototype alignment. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 12355–12364, 2020.

[48] Qiangeng Xu, Yin Zhou, Weiyue Wang, Charles R Qi, and Dragomir Anguelov. Spg: Unsupervised domain adaptation for 3d object detection via semantic point generation. In *IEEE International Conference on Computer Vision*, pages 15446–15456, 2021.

[49] Runsheng Xu, Yi Guo, Xu Han, Xin Xia, Hao Xiang, and Jiaqi Ma. Opencda: an open cooperative driving automation framework integrated with co-simulation. In *IEEE International Intelligent Transportation Systems Conference*, pages 1155–1162. IEEE, 2021.

[50] Runsheng Xu, Faezeh Tafazzoli, Li Zhang, Timo Rehfeld, Gunther Krehl, and Arunava Seal. Holistic grid fusion based stop line estimation. In *International Conference on Pattern Recognition*, pages 8400–8407. IEEE, 2021.

[51] Runsheng Xu, Hao Xiang, Xin Xia, Xu Han, Jinlong Li, and Jiaqi Ma. OPV2V: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication. In *IEEE International Conference on Robotics and Automation*, 2022.

[52] Jihan Yang, Shaoshuai Shi, Zhe Wang, Hongsheng Li, and Xiaojuan Qi. St3d: Self-training for unsupervised domain adaptation on 3d object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 10368–10378, 2021.

[53] Kaichao You, Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Universal domain adaptation. In *IEEE*

Conference on Computer Vision and Pattern Recognition, pages 2720–2729, 2019.

[54] Chong Zhang, Zongxian Li, Jingjing Liu, Peixi Peng, Qixiang Ye, Shijian Lu, Tiejun Huang, and Yonghong Tian. Self-guided adaptation: Progressive representation alignment for domain adaptive object detection. *IEEE Transactions on Multimedia*, 24:2246–2258, 2021.

[55] Yixin Zhang, Zilei Wang, and Yushi Mao. Rpn prototype alignment for domain adaptive object detector. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 12425–12434, 2021.

[56] Xiangmo Zhao, Pengpeng Sun, Zhigang Xu, Haigen Min, and Hongkai Yu. Fusion of 3d lidar and camera data for object detection in autonomous vehicle applications. *IEEE Sensors Journal*, 20(9):4901–4913, 2020.

[57] Zhen Zhao, Yuhong Guo, Haifeng Shen, and Jieping Ye. Adaptive object detection with dual multi-label prediction. In *European Conference on Computer Vision*, pages 54–69. Springer, 2020.

[58] Yangtao Zheng, Di Huang, Songtao Liu, and Yunhong Wang. Cross-domain object detection through coarse-to-fine feature adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 13766–13775, 2020.

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

[60] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision*, pages 2223–2232, 2017.

[61] Xinge Zhu, Jiangmiao Pang, Ceyuan Yang, Jianping Shi, and Dahua Lin. Adapting object detectors via selective cross-domain alignment. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 687–696, 2019.