applied sciences

MDPI

Article

An Object Detection Method Based on Feature Uncertainty Domain Adaptation for Autonomous Driving

Yuan Zhu [1], Ruidong Xu [1], Chongben Tao [2], Hao An [1], Zhipeng Sun [3] and Ke Lu [1,*]

[1] School of Automotive Studies, Tongji University, Shanghai 201800, China; yuan.zhu@tongji.edu.cn (Y.Z.); rd_xu@tongji.edu.cn (R.X.); anhao420@tongji.edu.cn (H.A.)
[2] The School of Electronic and Information Engineering, Suzhou University of Science and Technology, Suzhou 215009, China
[3] Nanchang Automotive Institute of Intelligence & New Energy, Tongji University, Nanchang 330013, China
* Correspondence: luke@tongji.edu.cn

Abstract: The environment perception algorithm in autonomous driving is trained in the source domain, leading to domain drift and reduced detection accuracy in the target domain due to shifts in background feature distribution. To address this issue, a domain adaptive object detection algorithm based on feature uncertainty is proposed, which can improve the detection performance of object detection algorithms in unlabeled data. Firstly, a local alignment module based on channel information is proposed, which can obtain the model's uncertainty about different domain data based on the feature channels obtained through the feature extraction network, achieving adaptive dynamic local alignment. Secondly, an instance-level alignment module guided by local feature uncertainty is proposed, which can obtain the corresponding instance-level uncertainty through ROI mapping. To improve the domain invariance of bounding box regression, a multi-class, multi-regression instance-level uncertainty alignment module is proposed, which can achieve spatial decoupling of classification and regression tasks, further improving the model's domain adaptive ability. Finally, the effectiveness of the proposed algorithm is validated on Cityscapes, KITTI, and real vehicle data.

Keywords: object detection; domain adaptation; uncertainty

check for updates

Citation: Zhu, Y.; Xu, R.; Tao, C.; An, H.; Sun, Z.; Lu, K. An Object Detection Method Based on Feature Uncertainty Domain Adaptation for Autonomous Driving. *Appl. Sci.* **2023**, *13*, 6448. https://doi.org/10.3390/app13116448

Academic Editor: Sungho Kim

Received: 1 May 2023
Revised: 17 May 2023
Accepted: 23 May 2023
Published: 25 May 2023

1. Introduction

Automatic driving vehicles require the accurate detection of surrounding objects to enable appropriate planning from subsequent decision-making algorithms. In recent years, deep-learning-based object detection algorithms [1–9] have achieved remarkable results, and many publicly available datasets have been used to evaluate algorithm performance. These models often rely on supervised training on large-scale datasets with ground truth annotations. However, compared to actual driving environments, the scenes contained in the datasets are mostly under good weather conditions. Training a detector based on such data (source domain) would result in the model learning only limited feature representations. When the scene changes, the pre-trained detector would suffer from a sharp decline in performance on data with different distributions (target domain). Therefore, improving the scene adaptation ability of the detector is an indispensable research topic in future automatic driving technology. Real-world traffic scenes present various difficulties in achieving accurate object detection due to different weather conditions, background styles, and target types. For example, rainy and foggy conditions can cause object occlusion, increased texture granularity, and blurred contours, making it difficult for the feature extraction layer of convolutional neural networks to obtain effective feature information, resulting in output bounding box offset and classification errors [10–12].

There are two different methods to solve this problem. The direct method involves continuously collecting data from various scenarios and obtaining corresponding labels through manual annotation. This approach is an attempt to combat infinite possibilities with limited resources and is not a sustainable route. Another method is to adopt domain adaptation to obtain domain-invariant features from unlabeled data or guide the model to learn features of the target domain [13]. The purpose of domain adaptation is to use both labeled source domain data and unlabeled target domain data to train the model. This process is usually subject to two mutually adversarial constraints, guiding the model to reduce the difference in feature distribution between domains. In most domain adaptation-based object detection methods, additional domain discriminators are added after the feature extraction stage and detection head to solve the domain drift problem through adversarial learning. However, directly applying domain discriminators on the feature maps obtained from the feature extraction layer for local alignment can cause changes to the already aligned features, which, in turn, affects the detection accuracy. For autonomous driving scenarios, changes in target features due to environmental variations are often local. As shown in Figure 1a, both foggy and clear images contain the same object. Although the targets in the foggy images appear blurry, most of their features are similar to those in the normal images. This means that the model should focus on aligning regions with significant feature changes. Therefore, we propose a feature uncertainty-based local alignment module, Feature Uncertainty Alignment (FUA), that can adaptively achieve feature alignment based on the local blurriness of an image. Figure 1b displays two feature channel entropy maps from different domains, where each pixel value represents the entropy of all channels at that location. A higher channel entropy indicates that the position contains richer feature information, which is advantageous for target detection. Conversely, when feature information is insufficient, the model needs to learn new feature representations to adapt to the target domain.

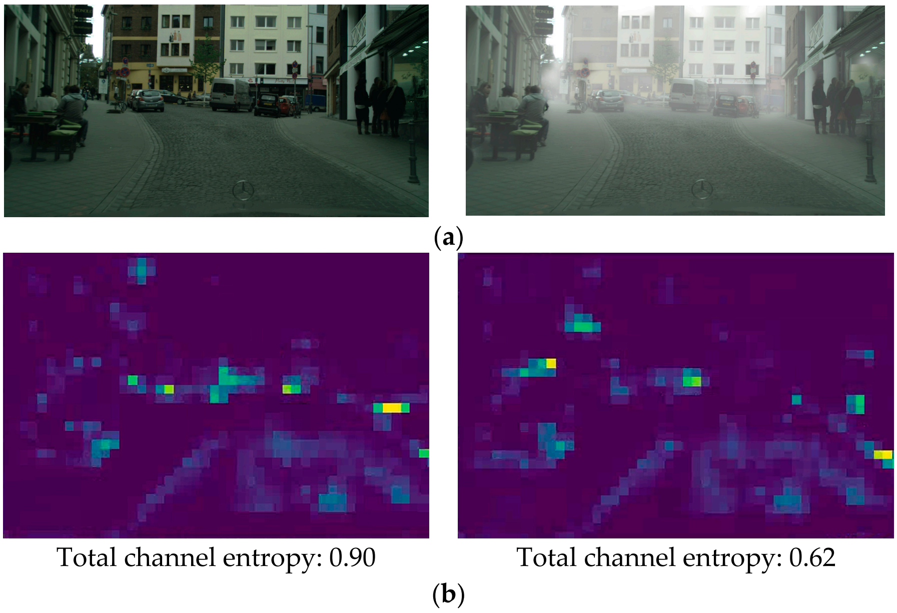


Figure 1. The influence of fog on image features. Each pixel value in the second row of images represents the feature channel entropy, which reflects the uncertainty of the feature extraction layer for the input data. A feature map of size D × H × W can obtain a 1 × H × W feature channel entropy map. (**a**) Normal image and foggy image. (**b**) Feature channel entropy map.

The methods based on adversarial learning have been widely developed in the fields of classification [14–19] and segmentation [20–23]. However, unlike classification and segmentation tasks, the commonly used global alignment methods ignore the decoupling of bounding box regression and object classification in object detection. When optimizing two different subtasks together, conflicts in the optimization space inevitably arise. A study [24] pointed out that the reason for this problem is that the decision boundaries in regression space and classification space are significantly different and proposed a domain adaptive module that utilizes auxiliary classifiers and locators to decouple the behavior inconsistency between the two tasks, thus improving the classification and localization ability of the detector. Based on this, this paper proposes a new instance-level uncertainty alignment (IUA) module guided by the aforementioned feature channel entropy. Moreover, a scale-independent entropy loss function is established for the domain adaptive loss of bounding box regression.

Specifically, the contributions of this paper are as follows:

(1) An object detection method based on feature uncertainty domain adaptation (FUDA) is proposed to address the problem of domain shift caused by environmental changes in autonomous driving object detection algorithms. By utilizing unlabeled target domain data to learn domain-invariant features, FUDA reduces cross-domain differences and improves the detection performance of the detector in different scenarios.

(2) To address the problem of feature degradation in the source domain caused by local domain alignment in current domain adaptation methods, a feature uncertainty-based local alignment module (FUA) is proposed, which can dynamically align features in low-entropy and high-entropy regions to improve the domain invariance in the extracted features by the backbone network in the target domain.

(3) An instance uncertainty alignment module (IUA) is proposed to address the problem of unstable alignment of bounding box regression in the target domain caused by existing global alignment methods. IUA guides the weight allocation of global alignment based on local uncertainty, which improves the accuracy of regression and classification in the target domain.

2. Related works

2.1. Adversarial Learning-Based Methods for Domain Adaptation

The widely adopted method in domain adaptation is to enable the model to learn domain-invariant features, where the features obtained from same-class objects from two different domains are similar. To achieve this goal, a classification task is coupled with an adversarial learning approach proposed in [13]. Specifically, a domain discriminator module is added after the output of the feature extraction layer, which is used to distinguish whether the current feature map comes from the source or target domain. During backpropagation, the gradient reversal layer is utilized to pass the opposite gradient, reducing the upper bound of the feature space, which requires the obtained features to meet the requirements of the classification task while also deceiving the domain discriminator. Inspired by this work, a study [25] attempted to apply this method in the field of object detection and proposed to add a domain discriminator in each stage of the two-stage detector, achieving local and global alignment through a gradient reversal layer. During backpropagation, both source and target domain data are used to calculate the domain discriminator loss, while the detection loss is obtained only from the source domain data and corresponding ground truth labels. A study [26] pointed out that the deep layers of the feature extraction layer tend to capture global information, and direct strong alignment is not the best choice. Therefore, they proposed to implement strong alignment only in the shallow layers of the backbone and weak alignment in the deeper layers. Meanwhile, the focal loss is introduced to distinguish the difficulty of recognizing targets, allocating different alignment weights. Many subsequent works [27–29] follow the above scheme and make improvements, but these methods calculate the domain difference using the entire feature map, causing the model to pay too much attention to task-irrelevant parts. To solve this problem, reference [30] utilized the ROIs generated by the RPN as the focus center and clustered the ROIs using the K-means algorithm. Different weights are assigned to the discriminator according to the selected regions. In addition to guiding attention through RPN, there are also works [31] that partition the importance of different regions by adding a semantic segmentation module. However, for rainy and foggy weather conditions in the autonomous driving scene, RPN or segmentation networks cannot provide perfect attention guidance. Therefore, in this paper, an uncertainty alignment module is proposed by calculating the entropy of the feature channel based on the different sensitivity of each channel to different regions in the feature extraction layer.

2.2. Methods Based on Image-to-Image Translation

Domain shift occurs when there is a difference in feature distribution between the source and target domains, caused by the fact that the input to the feature extraction layer is two visually distinct styles of images. Therefore, some works attempted to reduce the difference from the input end. A study [32] utilized Cycle-GAN [33] to learn a mapping function between the source and target domain images and added an image translation module in the input stage to improve the style consistency of the two images and improve the detection performance of the model in the target domain. Similarly, a study [34] proposed a progressive adaptation strategy that can enable a trained Cycle-GAN to achieve mutual mapping between the target and source domains, thereby reducing the feature difference between domains. Reference [35] points out the contradiction between feature discriminability and transferability in detectors. By generating synthetic samples of the target domain through a model translation module and then reweighting the data space based on importance, negative transfer issues are avoided. These methods are all searching for a perfect mapping function to transform images between different domains, allowing the detector to adapt to the target domain detection task without the need for retraining. However, this mapping function is not easy to obtain, and the detector has difficulty learning domain-invariant features.

2.3. Methods Based on Teacher–Student Model

Knowledge distillation is a commonly used method in semi-supervised learning. By adding perturbations to unlabeled samples and computing the consistency loss between the teacher and student models, it can improve the cross-domain robustness of the model [36]. Based on this, [37] proposed an unbiased average teacher model to deal with domain adaptation problems in object detection. This method trains a Cycle-GAN module to learn the image mapping between the source and target domains and then creates a target domain image that resembles the source domain and a source domain image that resembles the target domain. Knowledge distillation is achieved by matching the model predictions of the source-like target images with the original target domain images, and the teacher model's parameters are updated using the exponential moving average method. A study [38] also introduced an uncertainty module on top of the average teacher model to provide dynamically weighted parameters for the consistency loss. However, these methods globally unify the consistency difference between the student and teacher models, and their performance is easily affected by the added perturbations.

3. Domain Adaptation Object Detection Based on Feature Uncertainty

This paper proposes a domain adaptive object detection algorithm based on feature uncertainty. As shown in Figure 2, images from the source or target domain are downsampled, and semantic information is extracted using a backbone network. Similar to [25,26], FUA, a local image-level alignment method is employed, and a domain adaptive branch is added to reduce the distribution difference between the source and target domain features. However, the proposed FUA module calculates the channel entropy of the features to obtain the uncertainty of the current backbone for different regions. This result guides the domain adaptive module to focus more on areas with greater uncertainty. For global instance-level alignment, an IUA module is proposed, which contains multiple regression and classification outputs. The proposed regression entropy loss and classification entropy loss are used to obtain the adaptation of the detection head to the current feature map. Meanwhile, the introduced uncertainty map of FUA further guides the target of instance-level feature alignment.

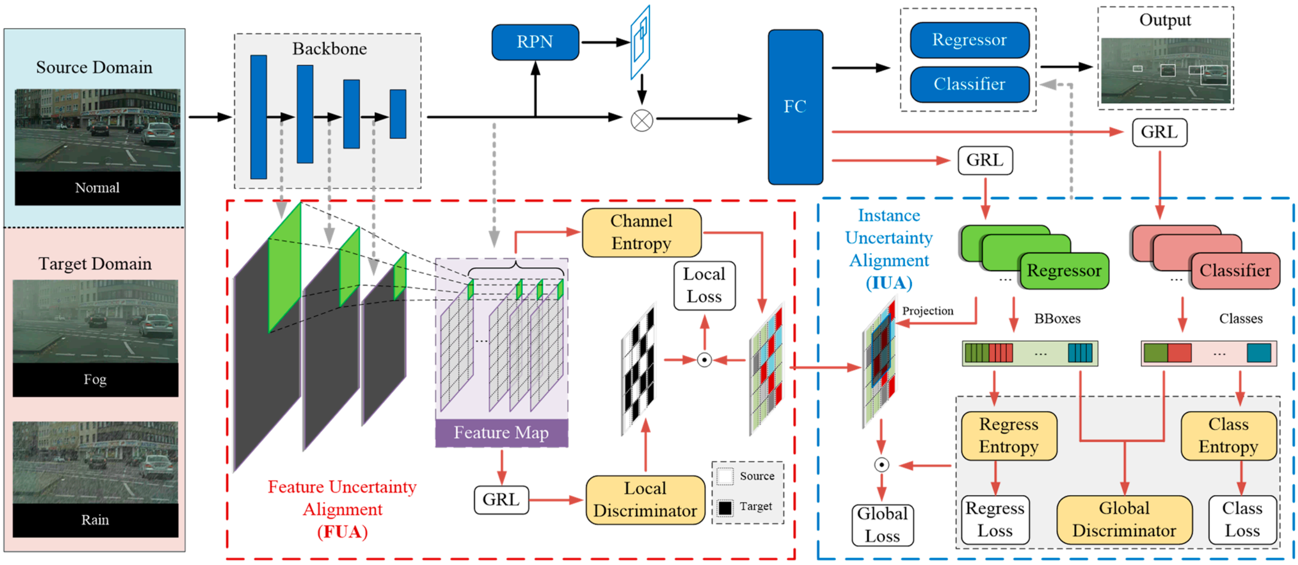


Figure 2. Diagram of proposed domain adaptive target detection algorithm based on feature uncertainty. The black arrows represent the data flow of the detector during the prediction phase. The red The gray arrows and red arrows represent the data flow of the detector during the train phase.

3.1. Problem Definition

Following the common representation of domain adaptation methods, the image from the source domain is represented as $X_S = \{x_1{}^S, x_2{}^S, \ldots, x_i{}^S\}, i = 1, 2, \ldots, n_S$, the label from the source domain is represented as $Y_S = \{y_1{}^S, y_2{}^S, \ldots, y_i{}^S\}, i = 1, 2, \ldots, n_S$, and the image from the target domain is represented as $X_T = \{x_1{}^T, x_2{}^T, \ldots, x_i{}^T\}, i = 1, 2, \ldots, n_T$. For the target detection task, the feature extraction network F extracts features from the input image to obtain a feature map, which is then fed to the RPN network R to generate numerous Region of Interest (ROI) proposals. Finally, the detection head H, consisting of multiple fully connected layers, produces the final object locations and classifications. The training loss in the source domain can be formulated as:

$$\mathcal{L}_{det} = \frac{1}{n_S} \sum_{i=1}^{n_S} H(R(F(x_i{}^S)), y_i{}^S) \quad (1)$$

For target domain data, similar loss objectives cannot be established due to the lack of labels $Y_T = \{y_1{}^T, y_2{}^T, \ldots, y_i{}^T\}$. Therefore, existing methods [25,26,29] attempt to add an additional local image-level domain discriminator D_{img} to guide $\mathrm{argmin}|F(x_i{}^S) - F(x_i{}^T)|$, which enables the network to produce similar feature maps for images from different domains. The local image-level domain discriminator D_{img} can distinguish whether the current feature is from the source domain or the target domain and achieve domain adaptation through adversarial training by using the gradient reversal layer (GRL). The training objective can be represented as:

$$\mathcal{L}_{img} = \frac{1}{n_S} \sum_{i=1}^{n_S} D_{img}(F(x_i{}^S)), l_i{}^S) + \frac{1}{n_T} \sum_{i=1}^{n_T} D_{img}(F(x_i{}^T)), l_i{}^T) \quad (2)$$

where $l_i{}^S$ and $l_i{}^T$ represent the classification labels for source and target domains, respectively, with 0 and 1. Similarly, the domain adaptive training objective for the detection head H can be expressed as:

$$\mathcal{L}_{ins} = \frac{1}{n_S} \sum_{i=1}^{n_S} D_{ins}(H(R(F(x_i{}^S))), l_i{}^S) + \frac{1}{n_T} \sum_{i=1}^{n_T} D_{ins}(H(R(F(x_i{}^T))), l_i{}^T) \quad (3)$$

where D_{ins} refers to instance-level domain discriminator. The final total loss can be expressed as:

$$\mathcal{L} = \mathcal{L}_{det} + \mathcal{L}_{img} + \mathcal{L}_{ins} \tag{4}$$

However, directly aligning the features may result in misalignment of the already aligned features. To address this issue, this paper proposes an uncertain alignment module based on feature channel entropy. In addition, existing methods are not suitable for the problem of target position and classification output in object detection algorithms. Therefore, an instance-level uncertainty alignment module based on multiple classifiers and regressors is proposed.

3.2. FUA

As shown in Figure 2, the feature maps of the source domain image X_S and the target domain image X_T obtained through the feature extraction network are input into a local discriminator D_{img}. This local discriminator needs to classify each pixel of the feature map and give the probability that the pixel is from the target domain. Conversely, the goal of local image-level alignment is to keep the source domain feature $F({x_i}^S)$ and the target domain feature $F({x_i}^T)$ obtained through the feature extraction network consistent so that the local discriminator D_{img} cannot identify whether the feature comes from the source domain or the target domain. To achieve this mutual adversarial goal, a gradient reversal layer (GRL) is added between the feature extraction network and the local discriminator, which acts as identity mapping during forward propagation and is used to reverse the gradient during backward propagation.

After being affected by rain and fog noise, not every region in the source domain image experiences significant feature distribution shift. Through training in the source domain, the feature extraction network can learn the unique feature representation of the source domain, resulting in more informative feature maps. The amount of information obtained in different regions with varying degrees of uncertainty also varies, which reflects the degree of domain shift between different regions in the image. To characterize these differences in the degree of shift, a feature channel entropy calculation method is proposed:

$$\mathcal{E}^C = -\sum_r O^C_{r,u,v} \cdot \log(O^C_{r,u,v}) \tag{5}$$

$$\begin{gathered}[O^C_{1,u,v}, O^C_{2,u,v}, \ldots, O^C_{R,u,v}] = \text{Softmax}(\mathbf{f_{r,u,v}}) \\ \mathbf{f_{r,u,v}} = [f_{1,u,v}, f_{2,u,v}, \ldots, f_{R,u,v}]\end{gathered} \tag{6}$$

where $f_{r,u,v}$ refers to the feature value at r_{th} (u,v) of feature map, $r \in [1,R], u \in [1,U]$, $v \in [1,V]$. In Figure 3, the feature channel entropy of the feature vector at each pixel position in the feature map represents the uncertainty of the corresponding features in the original image's receptive field. After training on the source domain, the feature extraction network will extract more effective information, resulting in more differentiation between the elements of the feature vector and a larger result for the feature channel entropy. Conversely, in the absence of training on target domain data, the feature channel entropy will be smaller due to the blurring of texture information in the entire image caused by rainy or foggy weather.

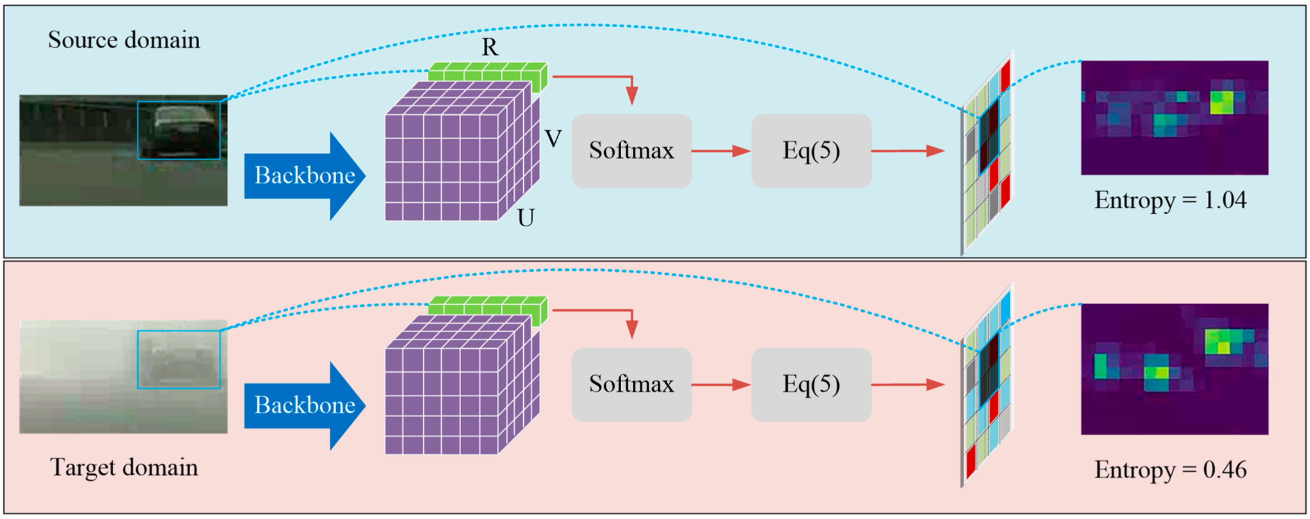


Figure 3. The difference in feature channel entropy in source domain and target domain images.

After calculating the feature channel entropy for each feature vector in the feature map of size (R × U × V) using Equation (5), a (1 × U × V) feature uncertainty map is obtained. This represents the uncertainty level of the current feature extraction network for different regions of the original image. Therefore, the local image-level domain adaptive loss function is given by:

$$\mathcal{L}_{img} = \sum_{u,v} \mathcal{E}^{C} \cdot [\mathcal{L}^{D}_{img}(p_{u,v}, l^{S}_{u,v}) + \mathcal{L}^{D}_{img}(p_{u,v}, l^{T}_{u,v})] \tag{7}$$

where ${l_{u,v}}^{S}$ and ${l_{u,v}}^{T}$ are 0 and 1, respectively, referring to the classification label of the source domain and target domain, $\mathcal{L}^{D}_{img}$ refers to the cross-entropy loss of the local image-level domain discriminator D_{img} and is defined as

$$\mathcal{L}^{D}_{img}(p_{u,v}, l_{u,v}) = -l_{u,v} \log(p_{u,v}) - (1 - l_{u,v}) \log(1 - p_{u,v}) \tag{8}$$

where $p_{u,v}$ refers to the domain classification result at the pixel location (u, v) of the feature map. By using Equation (8), a domain classification prediction map of size (1 × U × V) can be obtained. By incorporating image-level feature uncertainty into domain adaptive training through Equation (7), the feature extraction module can pay more attention to areas with larger domain gaps and reduce the weight of areas that are already well aligned.

3.3. IUA

The alignment at the image level is to deceive the domain discriminator by obtaining similar feature maps through the feature extraction network. Existing domain adaptation methods for global alignment [25,26,29] adopt a similar approach to image-level alignment by adding an instance-level domain discriminator to perform adversarial training on the classification and position output by the detection head. However, it was pointed out in [24] that this approach cannot adapt well to the task of object detection, as the learned transferable features were not decoupled for classification and regression tasks. The reason is that the classification task is discrete, while the regression task is continuous. From the training objective of object detection, stable detection results should be obtained for both source and target domain images. Therefore, we propose an instance-level uncertainty alignment based on multi-classifiers and multi-regressors, which introduces uncertainty to guide the instance-level alignment process while achieving the decoupling of classification and regression spaces.

A. Regression inconsistency

Domain adaptation tasks aim to capture domain inconsistencies across different stages of the model. As shown in Figure 4, for regression tasks, the coordinate points of the target bounding boxes lie in continuous space. To represent the inconsistency in the results obtained from the source and target domains, additional K parallel sub-regressors are proposed. Each sub-regressor uses the same network structure, and the randomness of each regressor is achieved through Dropout, resulting in K different sets of regression results. For objects from the source domain, the results of all sub-regressors should be close to the ground truth bounding box, with small differences between the coordinate points of the predicted bounding boxes. Conversely, objects from the target domain introduce more uncertainty, resulting in relatively large differences between the coordinate points of the bounding boxes. This difference represents the inconsistency of the model between the two domains. Therefore, reducing this inconsistency guides the model to adapt to the target differences between the two domains. A standard-deviation-based inconsistency loss was proposed in [24], but the standard deviation varies with the scale of the data. Under the same error ratio, the loss for small targets is smaller than that for large targets. This paper proposes a domain adaptive regression loss $\mathcal{L}_{rgs}^{En}$ based on region uncertainty:

$$\mathcal{L}_{En}^{rgs} = \frac{1}{4}\sum_{i=1}^{4} E_i^{rgs} \tag{9}$$

where E_i^{rgs} refers to the regression entropy of each parameter of the bounding box in K sub-regressors and represents the inconsistency of regression results, which can be expressed as:

$$E_i^{rgs} = -\sum_{j=1}^{K} P_j^{rgs} \log(P_j^{rgs}) \tag{10}$$

where P_j^{rgs} refers to the uncertainty of bounding box regression, ranging from 0 to 1, which can be expressed as:

$$P_j^{rgs} = \text{Softmax}(b_{ij}/\frac{1}{K}\sum_{j=1}^{K} b_{ij}), i = 1, \ldots, 4 \tag{11}$$

where b_{ij} refers to the i_{th} location parameter of bounding box output by j_{th} regressor.

B. Classification inconsistency

In order to capture the inconsistency between domains in the classification task, a similar approach to obtaining regression inconsistency is adopted by designing an additional N sub-classifiers. Each sub-classifier independently predicts the target class, which reflects the uncertainty of the model towards the input image in the classification task. Therefore, the classification inconsistency is addressed by utilizing an entropy-based domain adaptation loss. Assuming that there are C classes in the classification task, the classification loss $\mathcal{L}_{cls}^{En}$ can be formulated as:

$$\mathcal{L}_{cls}^{En} = \sum_{j=1}^{C} E_j^{cls}(\frac{1}{N}\sum_{i=1}^{N} o_{ij}) \tag{12}$$

where E_j^{cls} refers to the classification entropy of j_{th} classification of N sub-classifiers and represents the inconsistency of classification results, which can be expressed as:

$$E_j^{cls} = -\sum_{i=1}^{N} P_i^{cls} \log(P_i^{cls}) \tag{13}$$

where P_i^{cls} refers to the uncertainty of i_{th} classification, ranging from 0 to 1, which can be expressed as:

$$\begin{aligned} P_i^{cls} &= \text{Softmax}(\mathbf{o_{ij}}) \\ \mathbf{o_{ij}} &= [o_{1j}, o_{2j}, \ldots, o_{Nj}] \end{aligned} \tag{14}$$

where $\mathbf{o_{ij}}$ refers to the classification results of all N sub-classifiers in j_{th} classification.

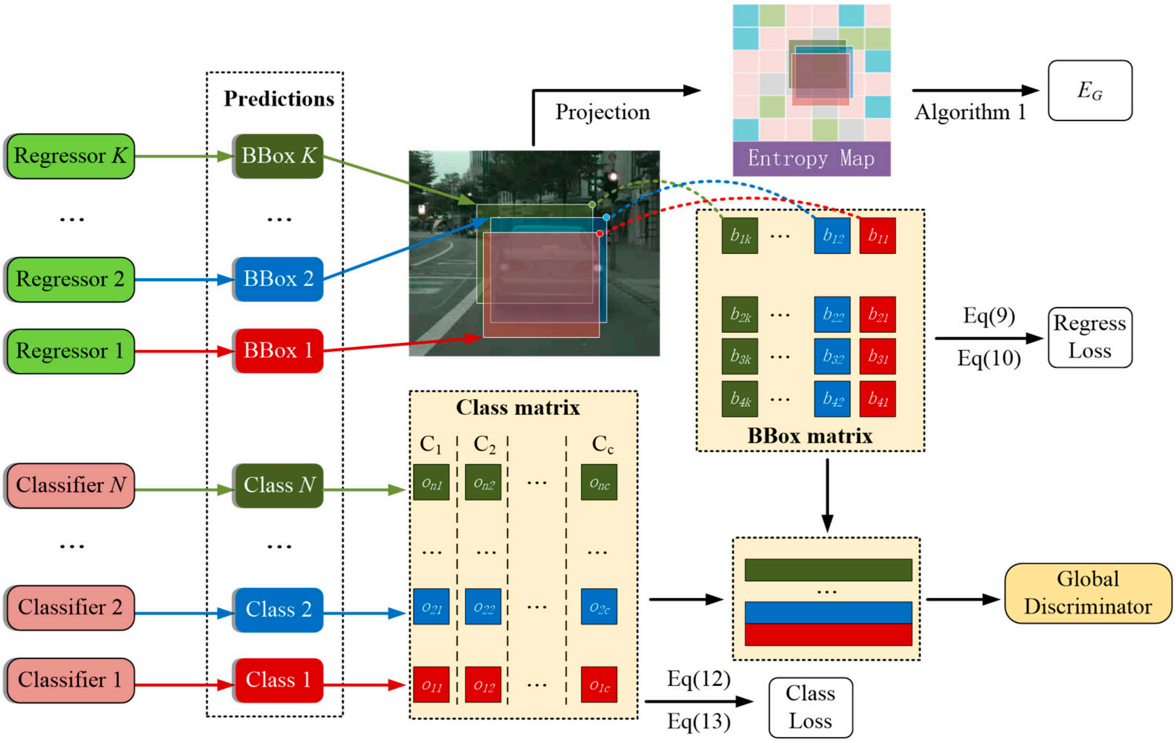


Figure 4. IUA module.

3.4. Instance-Level Uncertainty Guidance

The uncertainty caused by rainy and foggy weather was discussed in the FUA module, which also affects the position regression and classification of targets. Therefore, the feature uncertainty map was introduced into instance-level alignment. Specifically, assuming there are N_{ROI} ROIs, each ROI will obtain a mapping on the feature uncertainty map through Algorithm 1. Within each mapped region, there will be a different number of feature channel entropies. These values are pooled by averaging to obtain a representation of the entire ROI's uncertainty $\mathcal{E}_i^P, i \in [1, N_{ROI}]$. This uncertainty guidance can make the model pay more attention to targets with greater differences in instance-level alignment.

Algorithm 1 Feature Uncertainty Projection

```
Input: ROIs: $\mathbf{b_{ij}} = [b_{i1}, b_{i2}, b_{i3}, b_{i4}], i = 1, 2 \cdots N$
Output: Uncertainty of ROIs: $\mathcal{E}_i^P$ , $i = 1, 2 \cdots N$
Suppose The scale of downsampling by backbone is K
for $i = 1$ to $N$ by 1 do
    Calculate the projected box on uncertainty map $\mathbf{b_{ij}^P} = \mathbf{b_{ij}}/\text{K}$
    Obtain uncertainty of every ROI: $E_i^G = MeanPool\left(\mathbf{b_{ij}^P}\right)$
end for
return $\mathcal{E}_i^P$
```

3.5. Instance-Level Global Discriminator

The instance-level global domain discriminator can help reduce the differences in texture, size, and viewing angle of target domains. Therefore, a global domain discriminator is used to determine which domain each target's feature vector comes from. The train loss for this process is similar to that of the image-level domain discriminator and can be expressed as:

$$\mathcal{L}_{global}^{D} = \sum_{i=1}^{N_{ROI}} \left[-l_i \log(p_i) - (1 - l_i) \log(1 - p_i) \right] \tag{15}$$

where l_i is 0 or 1 and refers to the domain label of the source domain and target domain. The total loss of instance-level alignment can be expressed as:

$$\mathcal{L}_{ins} = \frac{1}{N_{ROI}} \sum_{i=1}^{N_{ROI}} \mathcal{E}_i^P [\alpha_1 \mathcal{L}_{rgs}^{En} + \alpha_2 \mathcal{L}_{cls}^{En} + \alpha_3 \mathcal{L}_{global}^{D}] \tag{16}$$

where $\alpha_1, \alpha_2, \alpha_3$ are the weight coefficients for balancing domain adaption loss. Therefore, the total train loss can be expressed as:

$$\mathcal{L}_{Total} = \eta_1 \mathcal{L}_{\det} + \eta_2 \mathcal{L}_{img} + \eta_3 \mathcal{L}_{ins} \tag{17}$$

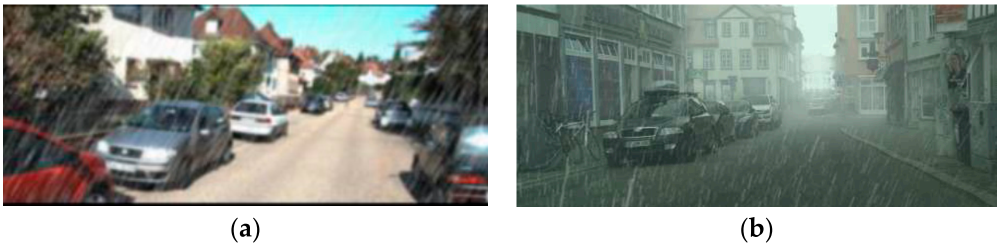
where η_1, η_2, η_3 are the weight coefficients for balancing total loss.

4. Experimental Analysis

4.1. Experimental Settings

During the training phase, each batch of data comprises one randomly selected image from the source domain and one from the target domain, which are used as inputs for the model. The source domain data include corresponding labels, whereas the target domain data do not. The baseline model FasterRCNN adopts ResNet-50 as the feature extraction layer and adds FPN as the neck layer before RPN. The proposed model is implemented based on Pytorch and MMDetection and trained and tested on an Ubuntu system with a single RTX 3090 GPU. The initial learning rate is set to 0.001, and the weight adjustment parameter is $\alpha_1 = 0.5, \alpha_2 = 0.5, \alpha_3 = 1, \eta_1 = 1, \eta_2 = 0.1, \eta_3 = 0.1$, updated using the stochastic gradient descent method with a momentum of 0.9 and a weight decay of 0.0001.

To evaluate the effectiveness of the proposed method, validation was performed on the Cityscapes foggy, Cityscapes rainy, KITTI rainy, and real-world test datasets. The Cityscapes dataset consists of 5000 traffic scene images in urban environments, including additional foggy scenes. The proposed model was trained with normal weather conditions and tested on foggy scenes to verify the effectiveness of domain adaptation. As the Cityscapes dataset lacks rainy weather scenes, a rainy noise generation algorithm was employed to generate the Cityscapes rainy dataset. The KITTI dataset contains 7481 traffic scene images, with 3712 images used for model training and 3769 images for model validation. KITTI is a commonly used dataset in autonomous driving research, but it only provides data under normal weather conditions. Therefore, similar to the Cityscapes rainy dataset, the KITTI rainy dataset was generated using a rainy noise generation algorithm. The established dataset of Cityscapes rainy and KITTI rainy are shown in the Figure 5.



(**a**) (**b**)

Figure 5. The established datasets of Cityscapes rainy and KITTI rainy. (**a**) KITTI rainy. (**b**) Cityscapes rainy.

4.2. Experimental Results of Cityscapes to Cityscapes Foggy

The impact of fog on images is characterized by significant local noise, which can randomly cause occlusion and blurring of the targets in some areas. Table 1 presents a comparison of detection results in Cityscapes foggy. These models only obtained ground truth labels from the source domain. When the basic FRCNN model was not adaptively trained for the target domain, only 20.0% of mAP is obtained in foggy scenes. In contrast, "Target" achieved a result of 42.3% by directly training on labels from foggy data. This result is only provided as a reference and is not included in the comparison because obtaining labels from the target domain is not feasible in practice. After adopting different domain adaptation methods, an obvious improvement in detection performance relative to FRCNN was observed. The proposed method achieved the highest detection results in multiple categories and had the highest mAP among all methods presented in Table 1.

Table 1. Experimental results (%) of Cityscapes to Cityscapes Foggy. Bold indicates the highest result.

Methods	Person	Rider	Car	Truck	Bus	Train	Motor	Bicycle	mAP
FRCNN [1]	24.2	28.5	34.6	10.1	25.6	4.0	11.7	20.9	20.0
DA [25]	25.0	31.0	40.5	22.1	35.3	20.2	27.1	20.0	27.6
SWDA [26]	29.9	42.3	43.5	24.5	36.2	32.6	30.0	35.3	34.3
MAF [28]	28.2	39.5	43.9	23.8	39.9	33.3	29.2	33.9	34.0
DAM [39]	30.8	40.5	44.3	27.2	38.4	34.5	28.4	32.2	34.6
CST [40]	32.7	44.4	50.1	21.7	**45.6**	25.4	30.1	36.8	35.9
CDN [41]	35.8	45.7	50.9	30.1	42.5	29.8	**30.8**	36.5	36.6
CFFA [42]	43.2	37.4	52.1	**34.7**	34.0	**46.9**	29.9	30.8	38.6
RPNPA [43]	33.3	45.6	50.5	30.4	43.6	42.0	29.7	36.8	39.0
FUDA (Ours)	**43.9**	**49.1**	**53.9**	28.4	41.8	31.6	28.2	**45.0**	**40.2**
Target	36.2	46.5	55.8	34.0	53.1	40.2	36.0	36.4	42.3

Compared with FRCNN, FUDA has the greatest improvement effect in the classification of Person, Rider, and Car and Bicycle, and the improvement in the mAP results reached 101%. It is worth noting that after training with the domain adaptation method, the results of several classifications exceed the results of training directly with the target domain labels. The reason is that under the constraint of adversarial training, the optimization objective of the feature extraction layer is opposite to that of the domain discriminator. In order to deceive the domain discriminator, the feature extraction layer will gradually make the output feature map uniform with the number of training iterations while meeting the target detection requirements. In other words, ideally when the feature distributions from the source and target domains are consistent, the domain discriminator will not be able to accurately judge the current domain classification. This indicates that the extracted features are domain-invariant, and the model obtained more generalized and universal target features.

The precision–recall curve of the proposed model in domain adaptation from Cityscapes to Cityscapes foggy is presented in Figure 6. Recall is the ratio of the number of correctly detected objects to all ground truth instances. Precision in Figure 6 indicates the percentage of correctly detected objects at different recall rates. It can be observed that the proposed model exhibits a clear advantage in the [0.2–0.6] interval.

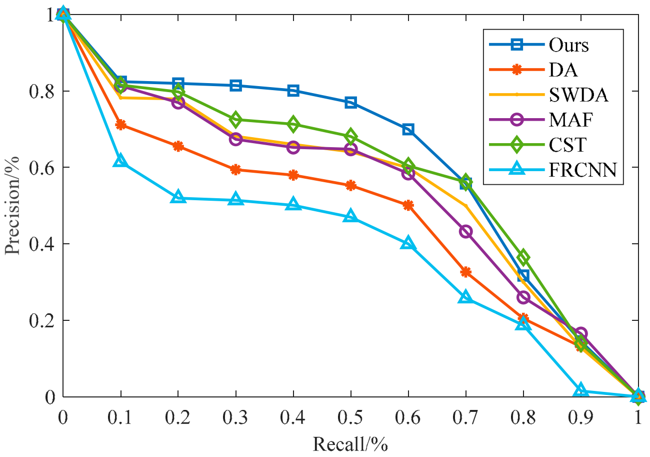


Figure 6. The PR curve of Cityscapes to Cityscapes foggy.

4.3. Experimental Results of Cityscapes to Cityscapes Rainy

The rainy weather introduces more random noise to the entire image compared to the foggy weather conditions. The experimental results for Cityscapes to Cityscapes rainy are presented in Table 2. It can be observed that DA, SWDA, and the proposed FUDA all achieved significant improvements over the baseline FRCNN model. FUDA obtained the highest detection results in multiple categories, and the mAP of 42.1% was the closest to the result obtained by directly training on the target domain data, which was 47.9%. The rainy weather conditions constructed in this study did not cause significant occlusion of the target objects, resulting in better overall experimental results compared to Cityscapes Foggy. After training with the FUDA method, the detection performance of FRCNN was improved by 81.5%.

Table 2. Experimental results (%) of Cityscapes to Cityscapes rainy. Bold indicates the highest result.

Methods	Person	Rider	Car	Truck	Bus	Train	Motor	Bicycle	mAP
FRCNN [1]	22.3	23.8	30.6	15.3	28.9	15.3	18.1	31.6	23.2
DA [25]	31.5	33.1	46.2	29.8	38.5	26.2	31.7	26.1	32.9
SWDA [26]	33.8	46.4	48.3	29.7	38.9	35.2	**38.2**	31.3	37.7
FUDA (Ours)	**38.4**	**48.2**	**56.9**	**35.1**	**42.8**	**39.6**	35.2	**40.3**	**42.1**
Target	44.2	53.8	58.3	41.7	49.6	45.7	42.4	47.2	47.9

4.4. Experimental Results of KITTI to KITTI Rainy

Table 3 presents the experimental results of KITTI to KITTI rainy. All models were trained on the original KITTI dataset and tested on the constructed KITTI rainy. The proposed FUDA achieved a remarkable mAP improvement of 51.8% compared to the baseline model FRCNN without domain adaptation training and a 4.3% improvement compared to SWDA.

Table 3. The experimental results (%) of KITTI to KITTI rainy. Bold indicates the highest result.

Methods	Car			Pedestrian			Cyclist			
	Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard	mAp
FRCNN [1]	58.9	34.7	32.0	33.5	26.3	19.6	43.2	39.4	31.8	35.5
DA [25]	62.7	55.1	40.8	39.6	25.1	24.3	56.2	50.1	48.4	44.7
SWDA [26]	69.1	60.7	51.9	**43.1**	**39.6**	31.3	62.4	56.0	50.8	51.7
Ours	**73.4**	**62.8**	**57.3**	40.3	35.3	**32.7**	**68.7**	**61.3**	**53.2**	**53.9**

4.5. Experimental Results of Phycical Testing Platform

As shown in Figure 7, in order to validate the proposed algorithm, we built a physical testing platform, which includes an 80-beam LiDAR, an RTK positioning system, a stereo camera, and a set of data acquisition devices. We collected two small-scale datasets in urban residential areas and urban streets. All sensor data were collected via ROS and synchronized to the same timestamp. The proposed algorithm was tested on camera data, which were converted into the format of the COCO dataset.

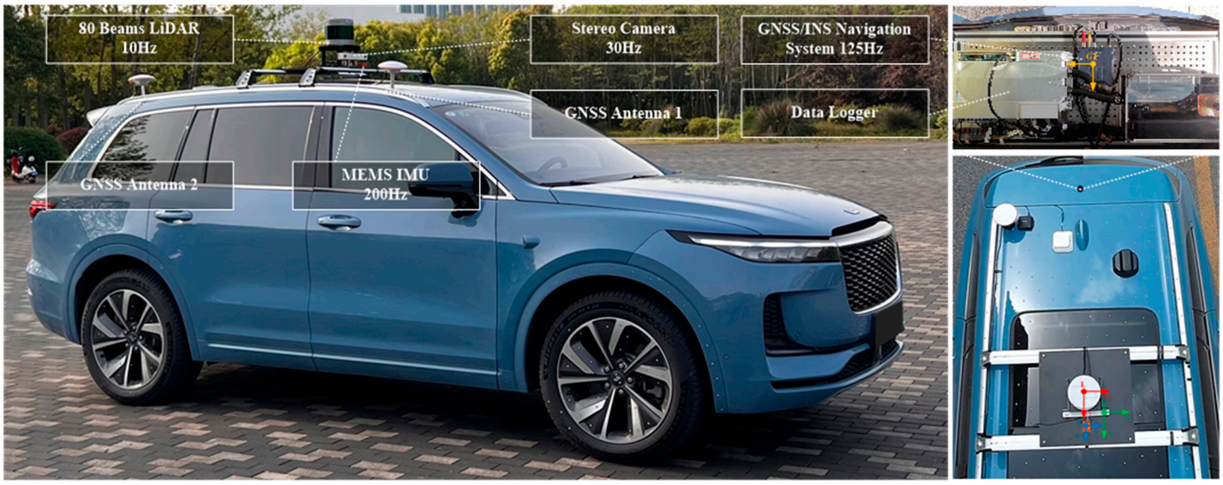


Figure 7. Physical testing platform.

The current publicly available datasets provide a lot of convenience for object detection research, but it is not easy to directly apply this to real-world scenarios. This is because the scene features, target categories, and weather conditions in public datasets cannot cover all real-world scenarios. This transfer process is also the problem that domain adaptation needs to solve. To further verify the effectiveness of the proposed model, we trained the model only with the labels of the Cityscapes dataset and then added some unlabeled real-world data as the target domain. Figure 8 shows the detection results of the original FRCNN without domain adaptation and the proposed FUDA in real-world data. It can be seen that the proposed method can correctly detect most of the targets, while FRCNN has many missed detections.

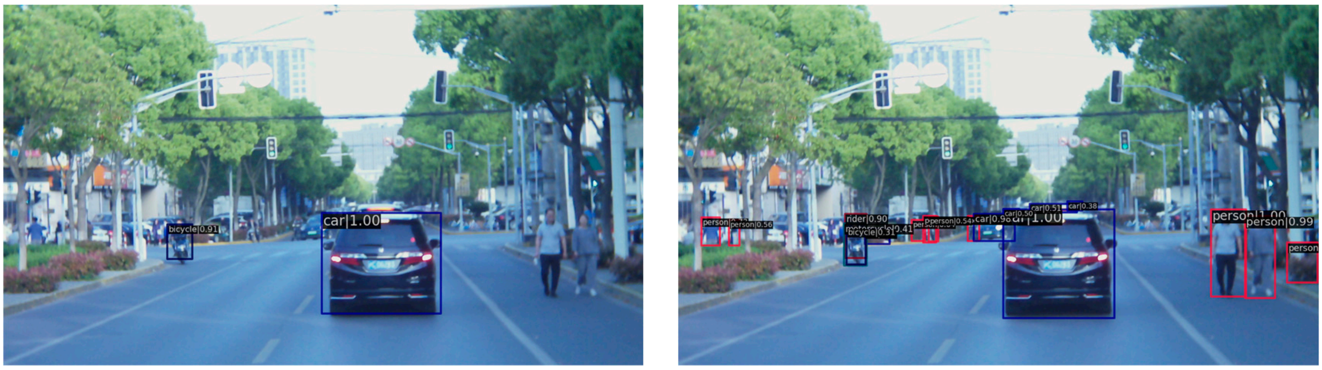
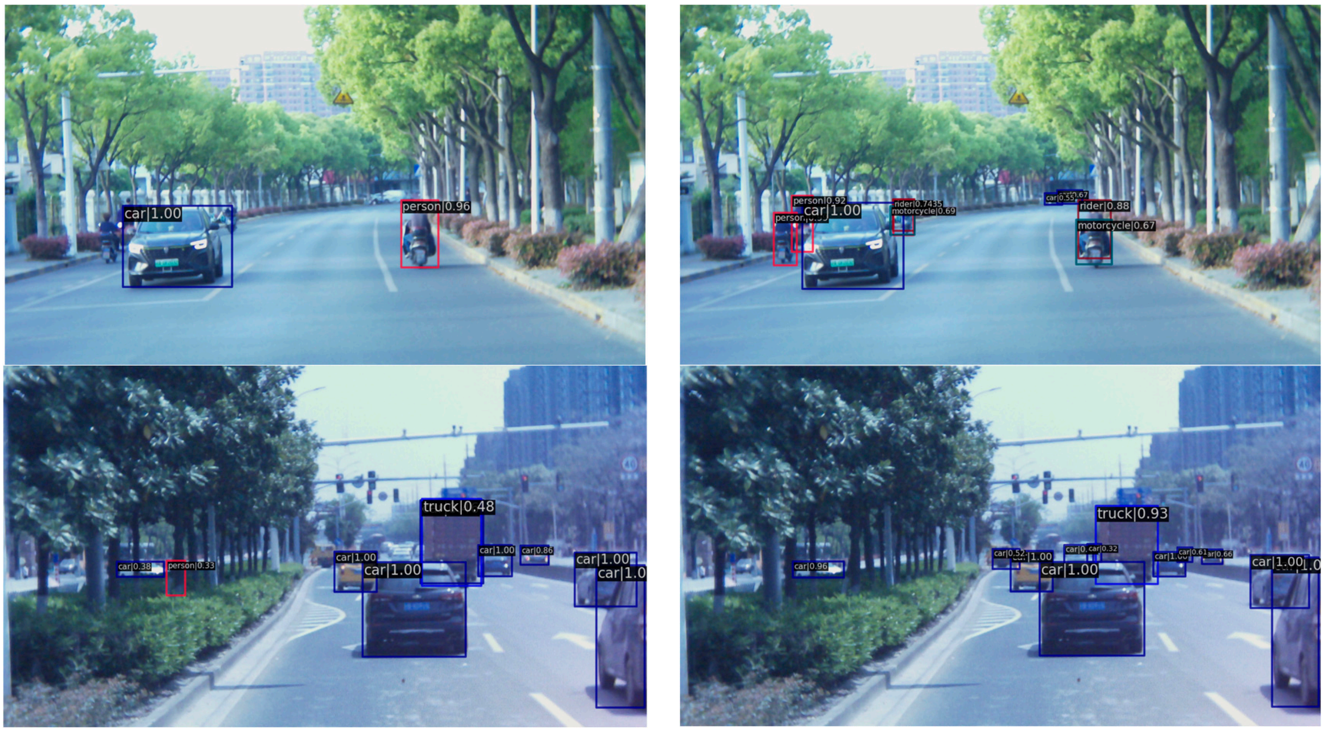


Figure 8. *Cont.*



FRCNN FUDA (Ours)

Figure 8. The visual results of Cityscapes to real data.

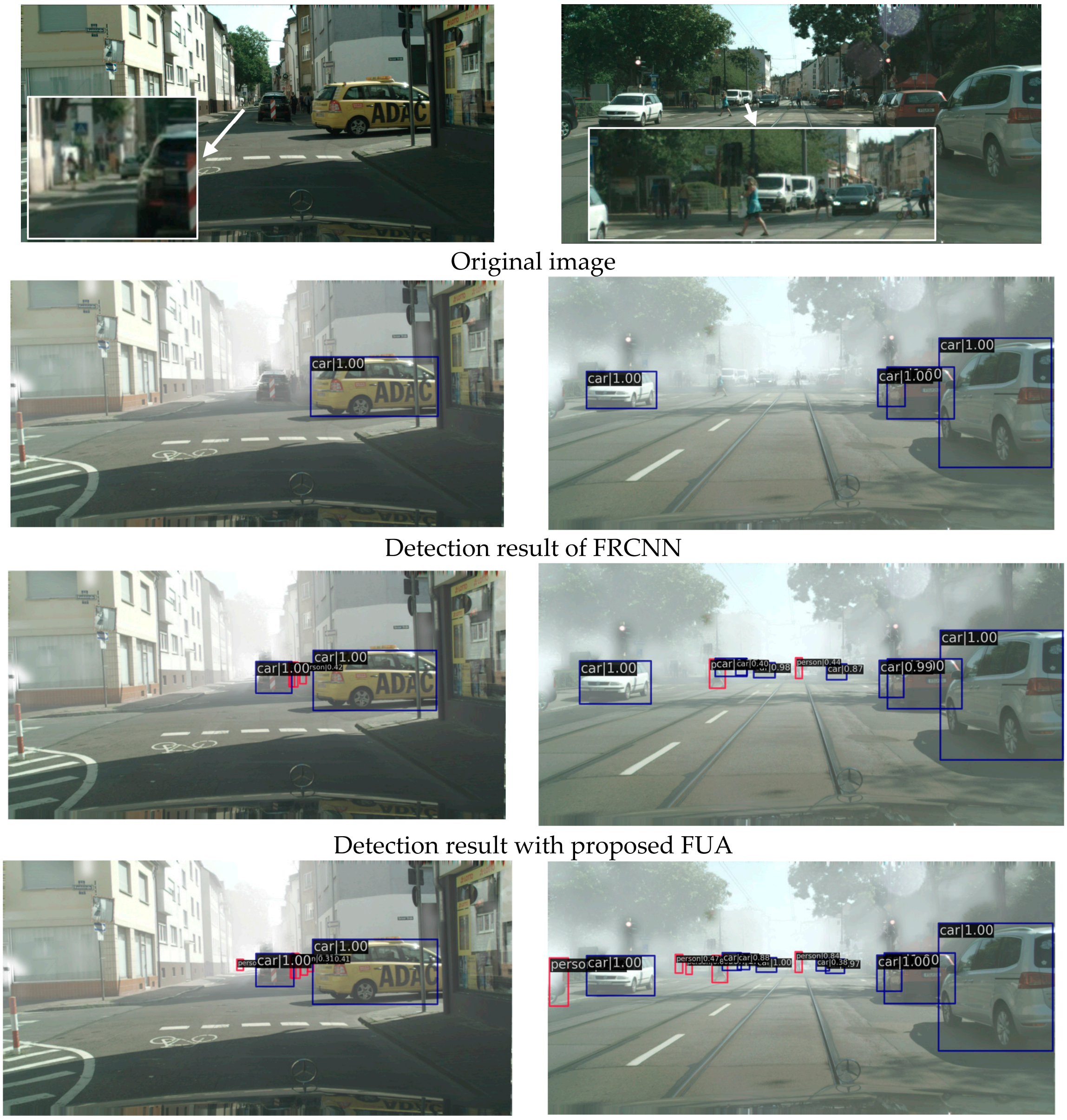
5. Ablation Study

The proposed FUDA leverages feature uncertainty to represent the degree of uncertainty of the feature extraction network on the current data feature distribution. This approach guides the domain adaptation process at the feature level by mapping the uncertainty maps to the instance-level alignment, enabling the model to assign different domain adaptation weights for each ROI during training. To analyze the effectiveness of the proposed FUA and IUA, Table 4 shows the performance improvements in different modules on detection. Without the addition of FUA and IUA modules, the proposed model achieves the same results as DA, which is used as a baseline for comparison. The results show that the addition of the FUA module leads to a 25.9% mAP improvement. Furthermore, with the simultaneous addition of FUA and IUA modules, the mAP improves by 44.8%. The FUA module contributes significantly to domain adaptation, while the IUA module further amplifies the performance improvement.

Table 4. Effects of FUA and IUA on model detection performance in Cityscapes to Cityscapes foggy. Bold indicates the highest result.

Methods	Person	Rider	Car	Truck	Bus	Train	Motor	Bicycle	mAP
DA	25.0	31.0	40.5	22.1	35.3	20.2	27.1	20.0	27.7
FUA	28.3	39.4	51.6	**28.8**	39.2	**32.4**	**31.5**	28.2	34.9
FUA + IUA	**43.9**	**49.1**	**53.9**	28.4	**41.8**	31.6	28.2	**45.0**	**40.2**

The visual results of Cityscapes to Cityscapes foggy are presented in Figure 9. It can be observed from Figure 9 that the model with the FUA module can detect vehicles and pedestrians that are affected by fog. Furthermore, the application of both the FUA and IUA modules enables the detection of distant targets. In contrast, the FRCNN model without domain adaptation can only detect large objects in close proximity. This demonstrates the effectiveness of the proposed FUA and IUA modules in providing domain adaptation.



Original image

Detection result of FRCNN

Detection result with proposed FUA

Detection result with proposed FUA and IUA

Figure 9. The visual results of Cityscapes to Cityscapes foggy.

5.1. *Effect on IUA Module*

To achieve the decoupling of feature space in classification and regression tasks, the IUA module adopts a multi-classifier and multi-regressor approach for instance-level alignment. To further analyze the effectiveness of the IUA module, we set different numbers of detection heads. Figure 10 shows the detection results of the model on Cityscapes to Cityscapes Foggy dataset with different numbers of heads. It can be observed that the model achieves the highest mAP when there are six heads, and the mAP result is lowest when no IUA module is added. This result is consistent with the findings in Section 5.1, indicating that while the FUA realizes feature-level alignment, instance-level alignment through detection heads is also important for the overall performance.

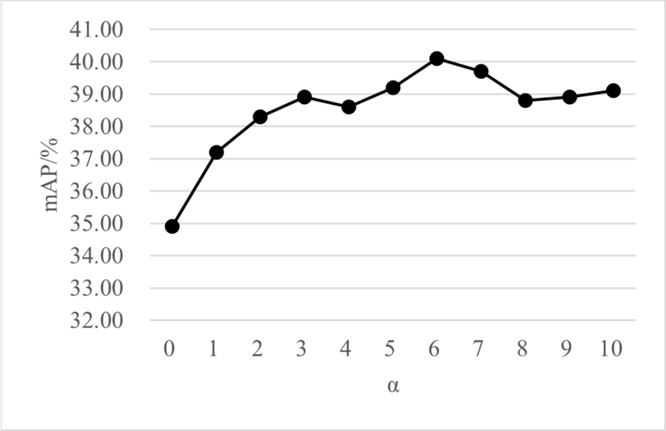


Figure 10. The impact of different detection heads in IUA on detection performance.

5.2. Visualization of Feature Distribution

The goal of domain adaptation is to ensure that the feature distributions obtained from different domains are similar. As shown in Figure 11, to analyze the effectiveness of the proposed domain adaptation method, t-SNE was used to reduce the dimensionality of the feature map output through the feature extraction network. The black squares represent results obtained from the source domain Cityscapes, while the red circles represent those obtained from the target domain Cityscapes foggy. It can be observed that without domain adaptation, the model extracts feature distributions that are further apart. This indicates that the model extracts different features from the source and target domain, making it difficult for the detection head to produce accurate results. After applying the proposed domain adaptation method, the model can extract similar feature distributions from two different domains, which improves its ability to obtain domain-invariant features from the target domain.

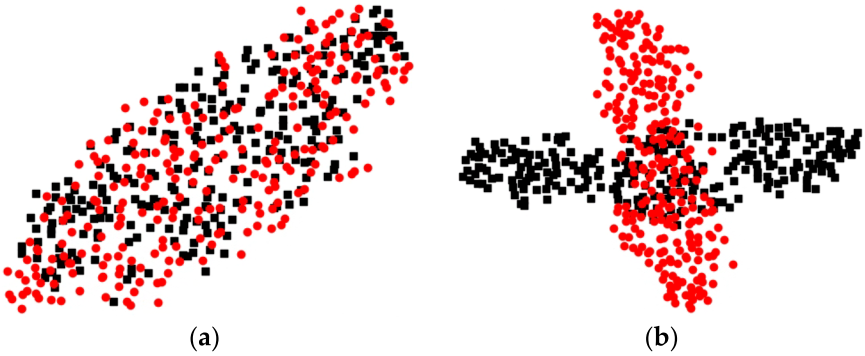


Figure 11. The visual result obtained from backbone via tSNE. (**a**) tSNE result with adaptive. (**b**) tSNE result without adaptive. The black squares represent results obtained from the source domain Cityscapes, while the red circles represent those obtained from the target domain Cityscapes foggy.

6. Conclusions

In this paper, we propose a feature uncertainty-based domain adaptive object detection algorithm to address the problem of domain shift caused by changes in background feature distributions in autonomous driving environments. Our method improves the detection performance of object detection algorithms in unlabeled data. To address the problem of source domain feature degradation caused by direct alignment between local domains in current domain adaptive methods, we propose a feature uncertainty-based local alignment module (FUA), which enhances the ability of feature extraction networks to acquire domain-invariant features in the target domain. We also propose an instance uncertainty alignment (IUA) module to address the unstable bounding box regression alignment issue that arises in current global alignment methods. Furthermore, this module enables spatial decoupling of classification and regression tasks, further enhancing the domain adaptation ability of the model. Finally, the effectiveness of our proposed algorithm is validated on Cityscapes, KITTI, and real data, achieving significant improvements in detection performance compared to baseline models. Test results on public datasets show that the proposed FUDA can enable the baseline model to effectively learn feature representations of target domains and achieve state-of-the-art results on multiple categories and mAP. The mAP results on Cityscapes Foggy, Cityscapes Rainy, and KITTI Rainy achieved 101%, 81.5%, and 51.8% improvements, respectively. The visualization results on real data further demonstrate the effectiveness of the proposed method. The results of ablation experiments specifically analyze the effect of the proposed FUA module and IUA module on detection performance.

Although our proposed domain adaptation method has shown the potential to enhance detection performance by leveraging unlabeled data, we acknowledge that the presence of extreme weather conditions poses a challenge, particularly when large quantities of target domain data are not readily accessible. In future research, we will concentrate on investigating the domain adaptation capability of our model using limited batches of target domain data. Furthermore, ensuring the stability of the domain adaptive training process is an aspect that deserves further examination and exploration.

Author Contributions: Conceptualization, Y.Z. and R.X.; methodology, Y.Z., R.X. and C.T.; formal analysis, H.A.; supervision, K.L.; writing—original draft, R.X.; writing—review and editing, C.T.; data curation, funding acquisition, Z.S. and K.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Perspective Study Funding of Nanchang Automotive Institute of Intelligence and New Energy, Tongji University (grant number: TPD-TC202211-07).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were analyzed in this study. These data can be found here: https://www.cvlibs.net/datasets/kitti/ (accessed on 23 May 2022) and https://www.cityscapes-dataset.com/downloads/ (accessed on 1 July 2022).

Acknowledgments: We appreciate the critical and constructive comments and suggestions from the reviewers that helped improve the quality of this manuscript. We would also like to offer our sincere thanks to those who participated in the data processing and provided constructive comments for this study.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1137–1149. [CrossRef] [PubMed]
2. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Springer: Berlin/Heidelberg, Germany, 2020; pp. 213–229.
3. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Online. 11–17 October 2021; IEEE: New York, NY, USA; pp. 10012–10022.
4. Tian, Z.; Shen, C.; Chen, H.; He, T. FCOS: Fully Convolutional One-Stage Object Detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; IEEE: New York, NY, USA, 2019; pp. 9627–9636.
5. Tao, C.; Cao, J.; Wang, C.; Zhang, Z.; Gao, Z. Pseudo-Mono for Monocular 3D Object Detection in Autonomous Driving. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, 1. [CrossRef]
6. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. CenterNet: Keypoint Triplets for Object Detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Long Beach, CA, USA, 27 October–2 November 2019; IEEE: New York, NY, USA, 2019; pp. 6569–6578.

7. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 21–26 July 2017; IEEE: New York, NY, USA, 2017; pp. 7263–7271.
8. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer: Cham, Switzerland, 2016; pp. 21–37.
9. Tao, C.; He, H.; Xu, F.; Cao, J. Stereo priori RCNN based car detection on point level for autonomous driving. *Knowl.-Based Syst.* **2021**, *229*, 107346. [CrossRef]
10. Hnewa, M.; Radha, H. Object Detection Under Rainy Conditions for Autonomous Vehicles: A Review of State-of-the-Art and Emerging Techniques. *IEEE Signal Process. Mag.* **2021**, *38*, 53–67. [CrossRef]
11. Rothmeier, T.; Huber, W. Performance Evaluation of Object Detection Algorithms Under Adverse Weather Conditions. In *Intelligent Transport Systems, from Research and Development to the Market Uptake*; Martins, A.L., Ferreira, J.C., Kocian, A., Costa, V., Eds.; Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering; Springer International Publishing: Cham, Switzerland, 2021; pp. 211–222. [CrossRef]
12. Hasirlioglu, S.; Riener, A. Challenges in Object Detection Under Rainy Weather Conditions. In *Intelligent Transport Systems, from Research and Development to the Market Uptake*; Ferreira, J.C., Martins, A.L., Monteiro, V., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 53–65. [CrossRef]
13. Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; Lempitsky, V. Domain-Adversarial Training of Neural Networks. *J. Mach. Learn. Res.* **2016**, *17*, 1–35.
14. Saito, K.; Ushiku, Y.; Harada, T. Asymmetric Tri-training for Unsupervised Domain Adaptation. In Proceedings of the 34th International Conference on Machine Learning, PMLR, Sydney, Australia, 6–11 August 2017; pp. 2988–2997. Available online: https://proceedings.mlr.press/v70/saito17a.html (accessed on 1 May 2022).
15. Kurmi, V.K.; Kumar, S.; Namboodiri, V.P. Attending to Discriminative Certainty for Domain Adaptation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–17 June 2019; IEEE: New York, NY, USA, 2019; pp. 491–500.
16. Saito, K.; Watanabe, K.; Ushiku, Y.; Harada, T. Maximum Classifier Discrepancy for Unsupervised Domain Adaptation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; IEEE: New York, NY, USA, 2018; pp. 3723–3732.
17. Saha, S.; Zhao, S.; Zhu, X.X. Multitarget Domain Adaptation for Remote Sensing Classification Using Graph Neural Network. *IEEE Geosci. Remote Sens. Lett.* **2022**, *19*, 1–5. [CrossRef]
18. Zhang, J.; Huang, J.; Tian, Z.; Lu, S. Spectral Unsupervised Domain Adaptation for Visual Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; IEEE: New York, NY, USA, 2022; pp. 9829–9840.
19. Shrivastava, A.; Shekhar, S.; Patel, V.M. Unsupervised Domain Adaptation Using Parallel Transport on Grassmann Manifold. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision, Steamboat Springs, CO, USA, 24–26 March 2014; pp. 277–284.
20. Chang, W.-L.; Wang, H.-P.; Peng, W.-H.; Chiu, W.-C. All About Structure: Adapting Structural Information Across Domains for Boosting Semantic Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; IEEE: New York, NY, USA, 2019; pp. 1900–1909.
21. Tsai, Y.-H.; Hung, W.-C.; Schulter, S.; Sohn, K.; Yang, M.-H.; Chandraker, M. Learning to Adapt Structured Output Space for Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; IEEE: New York, NY, USA, 2018; pp. 7472–7481.
22. Chen, Y.-H.; Chen, W.-Y.; Chen, Y.-T.; Tsai, B.-C.; Wang, Y.-C.F.; Sun, M. No More Discrimination: Cross City Adaptation of Road Scene Segmenters. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; IEEE: New York, NY, USA, 2017; pp. 1992–2001.
23. Zhang, P.; Zhang, B.; Zhang, T.; Chen, D.; Wang, Y.; Wen, F. Prototypical Pseudo Label Denoising and Target Structure Learning for Domain Adaptive Semantic Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; IEEE: New York, NY, USA, 2021; pp. 12414–12424.
24. Zhao, L.; Wang, L. Task-Specific Inconsistency Alignment for Domain Adaptive Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, NV, USA, 18–24 June 2022; IEEE: New York, NY, USA, 2022; pp. 14217–14226.
25. Chen, Y.; Li, W.; Sakaridis, C.; Dai, D.; Van Gool, L. Domain Adaptive Faster R-CNN for Object Detection in the Wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; IEEE: New York, NY, USA, 2018; pp. 3339–3348.
26. Saito, K.; Ushiku, Y.; Harada, T.; Saenko, K. Strong-Weak Distribution Alignment for Adaptive Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; IEEE: New York, NY, USA, 2019; pp. 6956–6965.
27. Zhang, S.; Tuo, H.; Hu, J.; Jing, Z. Domain Adaptive YOLO for One-Stage Cross-Domain Detection. In Proceedings of the 13th Asian Conference on Machine Learning, PMLR, Virtually, 17–19 November 2021; pp. 785–797.

28. He, Z.; Zhang, L. Multi-Adversarial Faster-RCNN for Unrestricted Object Detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; IEEE: New York, NY, USA, 2019; pp. 6668–6677.
29. Chen, Y.; Wang, H.; Li, W.; Sakaridis, C.; Dai, D.; Van Gool, L. Scale-Aware Domain Adaptive Faster R-CNN. *Int. J. Comput. Vis.* **2021**, *129*, 2223–2243. [CrossRef]
30. Zhu, X.; Pang, J.; Yang, C.; Shi, J.; Lin, D. Adapting Object Detectors via Selective Cross-Domain Alignment. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–17 June 2019; IEEE: New York, NY, USA, 2019; pp. 687–696.
31. Zhang, H.; Luo, G.; Li, J.; Wang, F.-Y. C2FDA: Coarse-to-Fine Domain Adaptation for Traffic Object Detection. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 12633–12647. [CrossRef]
32. Zhang, D.; Li, J.; Xiong, L.; Lin, L.; Ye, M.; Yang, S. Cycle-Consistent Domain Adaptive Faster RCNN. *IEEE Access* **2019**, *7*, 123903–123911. [CrossRef]
33. Zhu, J.-Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired Image-To-Image Translation Using Cycle-Consistent Adversarial Networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; IEEE: New York, NY, USA, 2017; pp. 2223–2232.
34. Hsu, H.-K.; Yao, C.H.; Tsai, Y.H.; Hung, W.C.; Tseng, H.Y.; Singh, M.; Yang, M.H. Progressive Domain Adaptation for Object Detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Snowmass Village, CO, USA, 1–5 March 2020; IEEE: New York, NY, USA, 2020; pp. 749–757.
35. Chen, C.; Zheng, Z.; Ding, X.; Huang, Y.; Dou, Q. Harmonizing Transferability and Discriminability for Adapting Object Detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; IEEE: New York, NY, USA, 2020; pp. 8869–8878.
36. Tarvainen, A.; Valpola, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2017.
37. Deng, J.; Li, W.; Chen, Y.; Duan, L. Unbiased Mean Teacher for Cross-Domain Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; IEEE: New York, NY, USA, 2021; pp. 4091–4101.
38. Zhou, Q.; Feng, Z.; Gu, Q.; Cheng, G.; Lu, X.; Shi, J.; Ma, L. Uncertainty-aware consistency regularization for cross-domain semantic segmentation. *Comput. Vis. Image Underst.* **2022**, *221*, 103448. [CrossRef]
39. Kim, T.; Jeong, M.; Kim, S.; Choi, S.; Kim, C. Diversify and Match: A Domain Adaptive Representation Learning Paradigm for Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–17 June 2019; IEEE: New York, NY, USA, 2019; pp. 12456–12465.
40. Zhao, G.; Li, G.; Xu, R.; Lin, L. Collaborative Training Between Region Proposal Localization and Classification for Domain Adaptive Object Detection. In *Computer Vision—ECCV 2020*; Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 86–102.
41. Su, P.; Wang, K.; Zeng, X.; Tang, S.; Chen, D.; Qiu, D.; Wang, X. Adapting Object Detectors with Conditional Domain Normalization. In *Computer Vision—ECCV 2020*; Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 403–419.
42. Zheng, Y.; Huang, D.; Liu, S.; Wang, Y. Cross-domain Object Detection through Coarse-to-Fine Feature Adaptation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; IEEE: New York, NY, USA, 2020; pp. 13766–13775.
43. Zhang, Y.; Wang, Z.; Mao, Y. RPN Prototype Alignment for Domain Adaptive Object Detector. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 12425–12434.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.