remote sensing

MDPI

Article

Rotation-Invariant and Relation-Aware Cross-Domain Adaptation Object Detection Network for Optical Remote Sensing Images

Ying Chen [1], Qi Liu [2,3], Teng Wang [4,*], Bin Wang [4] and Xiaoliang Meng [1]

1 School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China; jaycheney@whu.edu.cn (Y.C.); xmeng@whu.edu.cn (X.M.)
2 Westa College, Southwest University, Tiansheng Road No.2, Chongqing 400715, China; qliu24@utas.edu.au
3 College of Sciences and Engineering, University of Tasmania, Churchill Avenue, Hobart, TAS 7055, Australia
4 Surveying and Mapping Institute, Lands and Resource Department of Guangdong Province, Guangzhou 510500, China; icefirewb@foxmail.com
* Correspondence: wangteng43@hotmail.com; Tel.: +86-020-3833-4381

Abstract: In recent years, object detection has shown excellent results on a large number of annotated data, but when there is a discrepancy between the annotated data and the real test data, the performance of the trained object detection model is often degraded when it is directly transferred to the real test dataset. Compared with natural images, remote sensing images have great differences in appearance and quality. Traditional methods need to re-label all image data before interpretation, which will consume a lot of manpower and time. Therefore, it is of practical significance to study the Cross-Domain Adaptation Object Detection (CDAOD) of remote sensing images. To solve the above problems, our paper proposes a Rotation-Invariant and Relation-Aware (RIRA) CDAOD network. We trained the network at the image-level and the prototype-level based on a relation aware graph to align the feature distribution and added the rotation-invariant regularizer to deal with the rotation diversity. The Faster R-CNN network was adopted as the backbone framework of the network. We conducted experiments on two typical remote sensing building detection datasets, and set three domain adaptation scenarios: WHU 2012 → WHU 2016, Inria (Chicago) → Inria (Austin), and WHU 2012 → Inria (Austin). The results show that our method can effectively improve the detection effect in the target domain, and outperform competing methods by obtaining optimal results in all three scenarios.

Keywords: object detection; unsupervised domain adaptation; remote sensing images; rotation invariance; graph convolutional neural network (GCN)

Citation: Chen, Y.; Liu, Q.; Wang, T.; Wang, B.; Meng, X. Rotation-Invariant and Relation-Aware Cross-Domain Adaptation Object Detection Network for Optical Remote Sensing Images. *Remote Sens.* **2021**, *13*, 4386. https://doi.org/10.3390/rs13214386

Academic Editors: Gemine Vivone, Xinghua Li, Yongtao Yu, Xiaobin Guan and Ruitao Feng

Received: 25 August 2021
Accepted: 26 October 2021
Published: 30 October 2021

1. Introduction

Object detection is an important task in the field of computer vision [1–3]. It aims to predict the category and the precision position of the object in the image. In recent years, with the rapid development of deep convolutional neural networks [1], many excellent algorithms based on them have emerged in the field of object detection [4–8], and have achieved excellent performance on common datasets such as PASCAL VOC [9] and MS COCO [10]. However, these methods still face huge challenges in real life. When an object detection model trained on a dataset containing annotations is applied to a dataset not seen by other models, the performance of the model is often affected due to domain distribution discrepancy between the datasets. The most straightforward approach to this is to extensively collect other multi-source data and then annotate it to retrain the model, but, obviously, the time and labor costs are very high.

To reduce the discrepancy between domains among different datasets without additional annotations, researchers propose some unsupervised Cross-Domain adaptation

Object Detection (CDAOD) methods to transfer the information learned from a dataset containing a large number of annotations (source domain) to a dataset without annotations (target domain). In the field of computer vision, unsupervised CDAOD has been widely studied for classification [11–14] and segmentation [15–19]. However, the object detection task is different from these two problems, and the object detection task includes bounding-box localization and object classification, which makes the CDAOD problem more complex. The existing CDAOD methods can be roughly divided into the following four leading categories: (1) Adversarial feature learning [20,21], (2) Pseudo-label self-training [22,23], (3) Graph reasoning [24,25], and (4) Mean-teacher training [26]. However, the current research is centered on natural images, virtual game images, animation images, etc., mainly to solve the performance impact of the domain shift of different source images under illumination, weather, and style. There are few successful applications in the remote sensing field especially in the object detection task of remote sensing images. In the field of remote sensing, there are few studies on unsupervised CDAOD. To the best of our knowledge, recent works include the image-level domain transfer for multi-source remote sensing object detection by Li et al. [27], based on the generation adversarial network, and Koga et al. [28] proposed using the correlation alignment domain adaptation and adversarial domain adaptation to the region-based vehicle detector.

CDAOD for remote sensing images is a very challenging task in the field of remote sensing [29]. Most of the existing remote sensing open datasets come from Google images, but with the launch of China's satellite, there are more and more remote sensing image data of various types. Traditional methods need to re-label the multi-source data, which will consume a lot of manpower and time. Therefore, the research on CDAOD for multi-source remote sensing images has important application prospects and value in both military and civil fields. Due to the unique differences of optical remote sensing images in resolution and image forming mode, the domain adaptation object detection of optical remote sensing images is faced with unique challenges and constraints compared with natural images.

For the task of remote sensing the image CDAOD, because the remote sensing image is taken from an overlooking angle, compared with the natural image, the direction of the remote sensing image object in the image is arbitrary. However, the bounding box of our object detection task is horizontal. With the convolutional neural network [3], there is only translation invariance but no rotation invariance. Therefore, for remote sensing images, the rotation diversity of the object is a challenge to the detection performance. It is also very important to use inter-domain rotation invariance to better align objects of the same category in different domains.

On the other hand, remote sensing images taken from an overlooking angle are mostly ground objects with complex spectral texture features and high similarity with the detected objects. Although the background of natural image is complex, it is easy to distinguish from the detection object. In other words, the connection between objects and scenes in remote sensing images is usually closer than that of natural images, such as airplanes with airports, and ships with oceans. However, the region extracted by the common detection network can only reflect limited modal information, such as the specific scale and direction. Therefore, for the CDAOD of remote sensing images, we need to integrate the multi-modal information reflected by different instances into the prototype representation.

To solve these problems, we proposed a Rotation-Invariant and Relation-Aware (RIRA) CDAOD network of remote sensing images. This model can solve the above problems well, and our contribution can be summarized as follows:

- We propose a novel algorithm framework to solve the problem of unsupervised CDAOD in remote sensing images. The highest accuracy is achieved in three building detection adaptation scenarios with obvious domain shift.
- We propose to use the rotation-invariant regularizer term in both the source domain and the target domain to solve the problem of direction arbitrariness in remote sensing images.

- To aggregate regional information, a prototype-level domain alignment based on relation-aware graph is proposed to make the instance information obtained by the detection network more accurate.

2. Related Work

In this section, we will briefly review the most relevant work in the literature, including CDAOD in both the computer vision and remote sensing fields.

2.1. Cross-Domain Object Detection in the Computer Vision Field

In the community of computer vision, by analyzing the previous methods, we found that most of the previous studies used adversarial learning to align the feature distribution of the images in the source and target domains at the image-level and instance-level. The image-level refers to the shallow features such as the image background and scene layout. Instance-level refers to high-level semantic features such as appearance and pattern of objects. These methods eliminate domain shift in the feature extraction network of Faster R-CNN and the region-based classification regression network, respectively. Chen et al. [20] is the first work to propose the domain adaptation problem for the task of object detection. In their work, they employ adversarial feature learning to align domain distribution features at instance-level and image-level, respectively, and perform consistent regularization for domain classifiers. This paper also explains the effectiveness of adversarial learning from the perspective of probability theory. Saito et al. [21] argue that it is not necessarily optimal to align features separately at different levels. Therefore, they propose strong feature alignment at the local level and weak feature alignment at the global level for the previous image-level feature extraction part. With the rise of graph convolutional neural networks (GCN), the use of graphs to model relationships has become popular. Therefore, Cai et al. [24] proposed a mean-teacher Object Relation (MTOR) method, which used student–teacher training strategies to conduct model training and integrated graph reasoning. Recently, Xu et al. [25] proposed a method to derive prototype features based on graphs to achieve domain adaptation. This method aggregates information on RoI-Pooled proposals and uses a category balance loss function to optimize the domain alignment problem caused by category imbalance. Although these methods have achieved some success in natural images (e.g., Cityscapes [30]), virtual images (e.g., Sim10K [31]), and animation images (e.g., Watercolor [21]), they still cannot fully solve the problem of cross-domain object detection in remote sensing images because remote sensing images often show more complex structures.

2.2. Cross-Domain Object Detection in the Remote Sensing Field

In the field of remote sensing, there are few literature works on unsupervised CDAOD. Li et al. [27] evaluated the image-level domain adaptation tasks based on a popular generative adversarial network (GAN), AgGAN and CycleGAN, and performed cross-domain image segmentation based on U-Net and cross-domain object detection based on Faster R-CNN, respectively. Experiments show that almost all methods fail to improve the performance of Faster R-CNN in object detection, and the GAN-based method is the worst. In our paper, we will also compare our method with the image-level domain adaptation method based on AgGAN and CycleGAN. Koga et al. [28] applied correlation alignment (CORAL) domain adaptation and adversarial domain adaptation to a region-based vehicle detector, and the detection accuracy in the target region was improved by more than 10%. In that paper, the adversarial domain adaptation algorithm is further improved by using reconstruction loss to promote semantic feature learning. The accuracy of the proposed method is slightly better than that of the accuracy achieved with the labeled training data of the target domain. It can be seen from the two existing literature works that there is still a lack of in-depth research on unsupervised CDAOD in the field of remote sensing, and the experimental performance needs to be improved.

3. Methodology

This section presents the proposed Rotation-Invariant and Relation-Aware (RIRA) Cross-Domain adaptation Object Detection (CDAOD) network. We assume there are two domains from different sources and with different distributions. The source domain data have annotations for object detection, while the target domain does not. For the rest of our paper, we denote the source domain dataset as $D_{\mathcal{S}} = \{x_i^{\mathcal{S}}, b_i^{\mathcal{S}}, y_i^{\mathcal{S}}\}_{i=1}^{N_{\mathcal{S}}}$, where $N_{\mathcal{S}}$ is the total number of images in the source domain, $x_i^{\mathcal{S}}$ denotes the i-th image, $b_i^{\mathcal{S}}$ denotes the bounding box annotation and $y_i^{\mathcal{S}}$ denotes corresponding category label in the i-th source domain image. The total object category is denoted by C, so, including the background class, $y_i^{\mathcal{S}} \in \{1, 2, \ldots, C+1\}$. Furthermore, we set the target domain dataset as $D_{\mathcal{T}} = \{x_i^{\mathcal{T}}\}_{i=1}^{N_{\mathcal{T}}}$, where $N_{\mathcal{T}}$ is the total number of images in the target domain and $x_i^{\mathcal{T}}$ is the i-th target domain image.

The architecture of our proposed RIRA CDAOD network is illustrated in Figure 1. Our proposed rotation-invariant module achieves object rotation invariance by optimizing a new objective function, which explicitly applies rotation invariant regularization to force training samples before and after rotation to share similar features. For the image-level domain classifier network, we adopt the gradient reverse layer (GRL) [32] to minimize the domain classification loss of the domain classifier and maximize the classification loss of the feature extractor for adversarial training. For the prototype-level alignment network, in order to obtain class-based prototypes, we use RoI-based proposals to construct graphs with self-attention. Then, align at the prototype-level. Next, we will introduce each module of the network in detail.

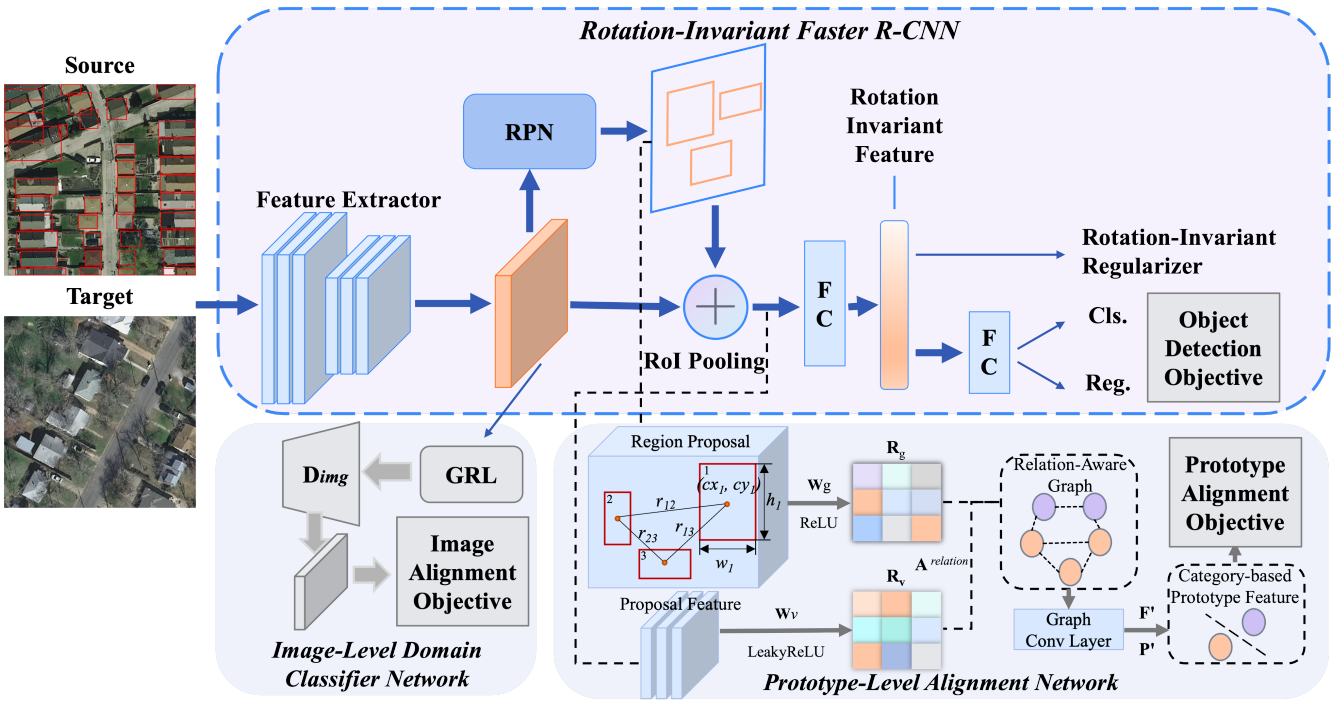


Figure 1. An overview of our Rotation-Invariant and Relation-Aware (RIRA) Cross-Domain adaptation Object Detection (CDAOD) network for optical remote sensing images.

3.1. Rotation-Invariant Regularizer

Faster R-CNN [7] uses the translational invariant "anchor" mechanism to solve the translational and scaling problems of the object, that is, for each position of the image, it predicts nine possible candidate windows (three scales and three aspect ratios), which are called anchors. The nine anchors are the same for any input image, so we only need to compute them once. This mechanism reduces the size of model parameters and reduces

the risk of overfitting on small datasets. The anchor mechanism could solve the scale invariance well; meanwhile, the convolution part can solve the translation invariance, but the Faster R-CNN network does not have the rotation invariance. For remote sensing images, the connection between the object and the scene is often greater than that between natural images, so it is of great significance to use rotation-invariant features to solve the direction arbitrariness of remote sensing images.

In the recent research work of remote sensing object detection [33,34], rotation invariance can significantly improve the detection effect for this problem. Through data augmenting, we do not need to modify any network structure on the original Faster R-CNN framework, but only need to redesign the objective function of the network. The objective function of the model is optimized by combining the rotation-invariant regularizer in our paper.

Given a set of labeled source domain samples $X_{\mathcal{S}}$, we generate a set of source training samples $\mathcal{X}_{\mathcal{S}} = \left\{X_{\mathcal{S}}, T_\phi X_{\mathcal{S}}\right\}$ by using a simple rotation operation, where $T_\phi = \left\{T_{\phi_1}, T_{\phi_2}, \cdots, T_{\phi_K}\right\}$ represents a family of K rotation transformations, where T_{ϕ_k} denotes the rotation operation of a training sample with the angle of $\phi_k \in \phi, k = 1, \cdots, K$.

In the Faster R-CNN framework, the only modification to Faster R-CNN is from the training of the object detection network (Fast R-CNN) [35]. In short, in addition to minimizing the classification loss and bounding box regression loss, we also added a rotation-invariant regularizer term to the CNN features to optimize the new objective function. We define the regularization constraint in the source domain as

$$\mathcal{L}_{rot}{}^{\mathcal{S}}\left(X_{\mathcal{S}}, T_\phi X_{\mathcal{S}}\right) = \frac{1}{2N_{\mathcal{S}}} \sum_{x_i^{\mathcal{S}} \in X_{\mathcal{S}}} \left\| \boldsymbol{O}_r\left(x_i{}^{\mathcal{S}}\right) - \overline{\boldsymbol{O}_r\left(T_\phi x_i{}^{\mathcal{S}}\right)} \right\|_2^2 \tag{1}$$

where $\|\cdot\|_2$ is the 2-norm of vectors; $\boldsymbol{O}_r(\cdot)$ denotes the output of the penultimate FC layer as shown in Figure 1; $\boldsymbol{O}_r\left(x_i{}^{\mathcal{S}}\right)$ is the CNN feature of the source training sample $x_i{}^{\mathcal{S}}$; $\overline{\boldsymbol{O}_r\left(T_\phi x_i{}^{\mathcal{S}}\right)}$ denotes the average CNN feature extracted from all rotated samples of the source training sample $x_i{}^{\mathcal{S}}$, and it is defined as

$$\overline{\boldsymbol{O}_r\left(T_\phi x_i{}^{\mathcal{S}}\right)} = \frac{1}{K} \sum_{j=1}^{K} \boldsymbol{O}_r\left(T_{\phi_j} x_i{}^{\mathcal{S}}\right) \tag{2}$$

For images in the target domain, although there are no annotations, the rotation-invariant feature design does not need labels, in the process of training, the source domain and the target domain share feature parameters. we could perform the same operation in the target domain. For the target domain, the optimization function is as follows:

$$\mathcal{L}_{rot}{}^{\mathcal{T}}\left(X_{\mathcal{T}}, T_\phi X_{\mathcal{T}}\right) = \frac{1}{2N_{\mathcal{T}}} \sum_{x_i^{\mathcal{T}} \in X_{\mathcal{T}}} \left\| \boldsymbol{O}_r\left(x_i{}^{\mathcal{T}}\right) - \overline{\boldsymbol{O}_r\left(T_\phi x_i{}^{\mathcal{T}}\right)} \right\|_2^2 \tag{3}$$

where $\boldsymbol{O}_r\left(x_i{}^{\mathcal{T}}\right)$ is the CNN feature of the target sample $x_i{}^{\mathcal{T}}$; $\overline{\boldsymbol{O}_r\left(T_\phi x_i{}^{\mathcal{T}}\right)}$ denotes the average CNN feature extracted from all rotated samples of the target sample $x_i{}^{\mathcal{T}}$.

With the source domain and target domain rotation-invariant regularizer to optimize the model, the operation can make the source domain and target domain easier to align. $L_{rot}{}^{\mathcal{S}}$ and $L_{rot}{}^{\mathcal{T}}$ are part of the overall objective function.

3.2. Prototype-Level Alignment Based on Relation-Aware Graph

In this section, we will show how to implement prototype-level domain alignment based on relation-aware graph. Based on the Region Proposal Network (RPN) [7] network, a relation-aware graph is generated, which uses self-attention to construct the adjacency matrix instead of the pre-defined adjacency matrix. After the graph is constructed, the features of the proposal are aggregated according to the category to form the prototype features. Prototype-level features are then used for domain distribution alignment.

3.2.1. Region Proposal Generation

The RPN network is used to generate region proposals. The network judges that anchors belong to foreground or background through softmax, and then uses bounding box regression to correct anchors to obtain rough proposals. These proposals provide a wealth of information such as the scale and scene style of the instance, but due to the deviation of the bounding box, especially in the target domain, it often contains incomplete instance information. Subsequent operations aim to extract more precise information about instances from the region proposal through information aggregation. The details of RPN's network structure are shown in Figure 2.

3.2.2. Constructing Relation-Aware Graph

We built our relation-aware graph on the proposals extracted by RPN. Consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ corresponds to the set of nodes proposed by N_p, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ represents the set of edges. A self-adapted adjacency matrix $\mathbf{A}^{relation} \in \mathbb{R}^{N_p \times N_p}$ is used to model the relations between proposals by performing self-attention. To better model the relations between proposals, we refer to [36] to model the graph between objects respectively from the geometric relationship and visual relationship, to obtain the $\mathbf{A}^{relation}$. Next, we will define the geometric and visual relationships.

Geometric relationship. The geometric relationship module models the spatial relationship between two proposals by calculating geometric relationship features. This module takes the relevant geometric features of the proposal as input, projects them into a subspace, and measures the correlation between them by multiplying them by the geometric weight $\mathbf{W}_g \in \mathbb{R}^{d_g \times 8}$, where d_g is the dimension of the geometric relationship feature. The formula is as follows:

$$R_{ij}^{g} = \mathrm{ReLU}\left(\mathbf{W}_g r_{ij}^{g}\right) \tag{4}$$

In order to prevent the geometric relationship from being affected by scale and shifting, we define the relevant geometric features of the two region proposals as follows:

$$r_{ij}^{g} = \left[w_i, h_i, w_j, h_j, \frac{\|cx_i - cx_j\|}{w_j}, \frac{\|cy_i - cy_j\|}{h_j}, \log\left(\frac{w_i}{w_j}\right), \log\left(\frac{h_i}{h_j}\right)\right]^{\mathrm{T}} \tag{5}$$

where w_i and h_i are the width and height of the i-th region proposal, respectively, normalized by the image scale, so that $0 < w_i, h_i < 1$ and $1 \leqslant i, j \leqslant N_p$. (cx_i, cy_i) is the location of the center point of the i-th proposal. ReLU is used to truncate the feature response to zero.

Visual relationship. It is also necessary to explore the visual relationship between the two region proposals, that is, the importance of the features of the node j to the node i. The relationship can be learned adaptatively by self-attention without pre-defining the adjacency matrix. The module input is a set of proposal features, $\mathbf{F} = \left\{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_{N_p}\right\} \in \mathbb{R}^{N_p \times d}$, where d indicates the dimension of node features.

We concatenate the features of the two RoIs to obtain a sufficient representation to describe the relationship between the two RoIs, and convert the obtained features to another high-dimensional space. To this end, a learnable and shareable transformation matrix $\mathbf{W}_v \in \mathbb{R}^{d_v \times 2d}$ is applied to each pair of RoI features, where d_v is the dimension of visual relation features. Then, it can be calculated as follows:

$$R_{ij}^{v} = \mathrm{LeakyReLU}\left(\mathbf{W}_v\left[\mathbf{f}_i \parallel \mathbf{f}_j\right]\right) \tag{6}$$

where $\parallel$ denotes the concatenation operation.

Joint geometric and visual relations. The object relationship coefficient is calculated as the weighted sum of geometric and visual relationship features, as shown below:

$$A_{ij}^{relation} = \frac{R_{ij}^{g} \cdot \exp\left(R_{ij}^{v}\right)}{\sum_m R_{mj}^{g} \cdot \exp\left(R_{mj}^{v}\right)} \tag{7}$$

Given the object relationship coefficient $A_{ij}^{relation}$ between each pair of nodes, we can form feature aggregation between RoIs through the Relation-Aware Graph Convolutional layer, which allows each node to be affected by other nodes, based on their object relationship.

3.2.3. Graph-Based Region Aggregation

We obtained the self-adapted adjacency matrix $\mathbf{A}^{relation}$, from which we can obtain the normalized adjacency matrix $\tilde{\mathbf{A}}^{relation}$. It can be calculated as follows:

$$\tilde{A}_{ij}^{relation} = \frac{\exp\left(A_{ij}^{relation}\right)}{\sum_k \exp\left(A_{ik}^{relation}\right)} \tag{8}$$

Next, we use GCN [37] to aggregate the information from the graph. Unlike standard convolutions that operate on local Euclidean structures in an image, the goal of GCN is to learn a function $f(\cdot,\cdot)$ on a graph $\mathcal{G}$, which takes feature descriptions $\mathbf{F}^l \in \mathbb{R}^{N_p \times d}$ and a self-adapted adjacency matrix $\mathbf{A}^{relation}$ as inputs, and updates the node features as $\mathbf{F}^{l+1} \in \mathbb{R}^{N_p \times d'}$. Every GCN layer can be written as a nonlinear function by

$$\mathbf{F}^{l+1} = f\left(\mathbf{F}^l, \mathbf{A}^{relation}\right) \tag{9}$$

Employing the convolutional operation of $f(\cdot,\cdot)$ can be represented as

$$\mathbf{F}^{l+1} = \sigma\left(\tilde{\mathbf{A}}^{relation}\mathbf{F}^l\Theta^l\right) \tag{10}$$

where $\Theta^l \in \mathbb{R}^{d \times d'}$ is a transformation matrix to be learned and $\sigma(\cdot)$ denotes activation function, which is activated by LeakyReLU in our experiments. For each RoI-pooled feature corresponding to the classification probability vector $\mathbf{p} \in [0,1]^{1 \times C}$ and probability matrix $\mathbf{P} \in [0,1]^{N_p \times d}$, the information propagation process can be written as follows:

$$\mathbf{P}^{l+1} = \sigma\left(\tilde{\mathbf{A}}^{relation}\mathbf{P}^l\Theta^l\right) \tag{11}$$

Thus, we can learn and model the accurate relation of the proposals by stacking multiple GCN layers. In Equations (10) and (11), after region aggregation, $\mathbf{F}^{l+1} \in \mathbb{R}^{N_p \times d'}$ and $\mathbf{P}^{l+1} \in [0,1]^{N_p \times d'}$ express more accurate instance-level information through information propagation among neighboring proposals. The algorithm flow of how the relation-aware graph convolution layer is constructed is shown as Algorithm 1.

3.2.4. Generating Prototype Features

Using the spatial correlation provided by self-adapted adjacency matrix $\mathbf{A}^{relation}$, proposals' feature descriptions $\mathbf{F}^{l+1}$ and probability matrix $\mathbf{P}^{l+1}$ are aggregated. Now that the feature representation is aggregated at the instance-level, we want to integrate information reflected by the different instances into the prototype representation. To get the prototype features, we use the following formula:

$$\mathbf{v}_c = \frac{\sum_{i=1}^{N_p} \mathbf{P}_{ic}^{l+1} \cdot \left[\mathbf{F}_i^{l+1}\right]^{\mathcal{T}}}{\sum_{i=1}^{N_p} \mathbf{P}_{ic}^{l+1}} \tag{12}$$

where $\mathbf{v}_c \in \mathbb{R}^{1\times d'}$ denotes the prototype feature of class c. The prototype features generated here will be used for subsequent domain alignment based on prototype features, producing category-based prototype features in the source domain and target, respectively. Aligning prototype features is performed by minimizing the distance between prototypes in different domains of the same category and maximizing the distance between prototypes in different categories. The derived prototypes serve as the proxy of each class during subsequent domain alignment.

Algorithm 1: Relation-Aware Graph Convolutional Layer.
The number of the layer is L. Geometric transformation matrix: $\mathbf{W}_g^l \in \mathbb{R}^{d_g}$. Visual transformation matrix: $\mathbf{W}_v^l \in \mathbb{R}^{d_v}$. The number of RoIs is N_p. Layer transformation matrix: $\Theta^l \in \mathbb{R}^{d_{in}\times d_{out}}$.

Input: Input proposal features $\mathbf{F}^l \in \mathbb{R}^{N_p\times d}$.
Probability matrix $\mathbf{P}^l \in [0,1]^{N_b} \times d$.
RoIs position $\{w_i, h_i, cx_i, cy_i\}_{i=1,\cdots,N_p}$.
Output: $\mathbf{F}^L$ and $\mathbf{P}^L$.
for $l = 1$ **to** L **do**
a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
Geometric relationship:
Calculate $\mathbf{R}_g \in \mathbb{R}^{N_p\times d_g}$ using Equation (4)
Visual relationship:
Calculate $\mathbf{R}_v \in \mathbb{R}^{N_p\times d_v}$ using Equation (6)
Joint geometric and visual relations:
Calculate $\mathbf{A}^{relation} \in \mathbb{R}^{N_p\times N_p}$ by Equation (7)
Calculate $\tilde{\mathbf{A}}^{relation} \in \mathbb{R}^{N_p\times N_p}$ by Equation (8)
Graph-based region aggregation:
Calculate $\mathbf{F}^{l+1} \in \mathbb{R}^{N_p\times d'}$ and $\mathbf{P}^{l+1} \in [0,1]^{N_p\times d'}$ by Equations (10) and (11)
end

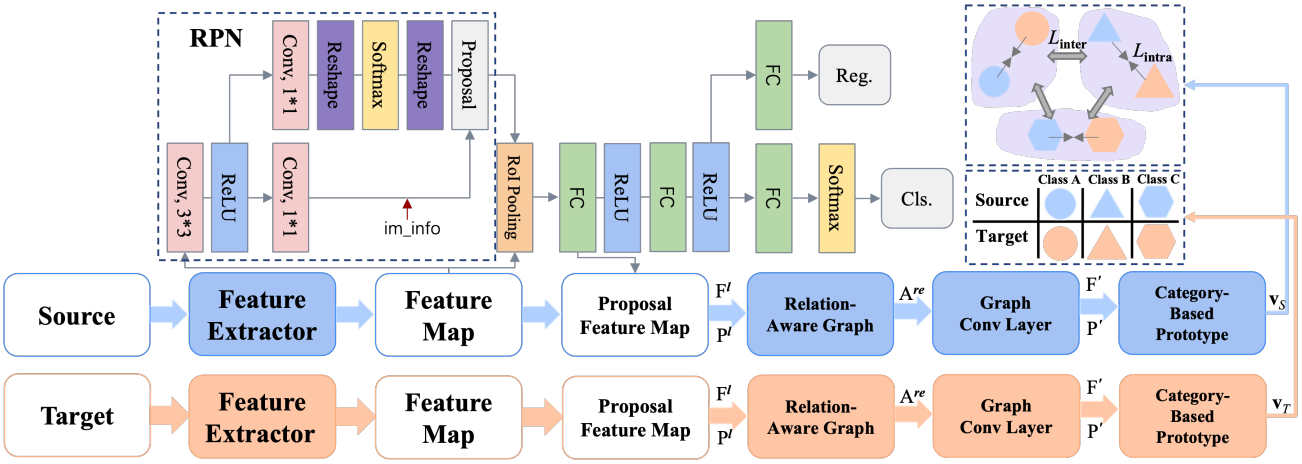


Figure 2. RPN's network structure and visual representation of prototype alignment.

3.2.5. Category-Level Domain Alignment

To reduce domain discrepancy at the prototype-level, we reduced the intra-class differences between the source and the target domain. Narrowing the distance between the same class prototype of two domains is done by minimizing the intra-class loss, denoted as $\mathcal{L}_{intra}$. The definition of $\mathcal{L}_{intra}$ is as follows:

$$\mathcal{L}_{intra}(D_\mathcal{S}, D_\mathcal{T}) = \frac{\sum_{i=1}^{C+1} \left\| \mathbf{v}_i^\mathcal{S} - \mathbf{v}_i^\mathcal{T} \right\|_2}{C+1} \tag{13}$$

In addition, we propose that the distances between prototypes of different classes should also be restricted by an inter-class loss, namely $\mathcal{L}_{inter}$. In the inter-class loss, the distance between prototypes of different classes is limited to larger than a margin. The formula is defined as follows:

$$\mathcal{L}_{inter}(D, D') = \frac{1}{C}\Big(\sum_{1 \leqslant i \neq j \leqslant C+1} max\Big(0, m - \Big\|\mathbf{v}_i^D - \mathbf{v}_i^{D'}\Big\|_2\Big)\Big) \tag{14}$$

where D and D' represent two domains from which prototype pairs belonging to different categories can be selected. D and D' can be the same domain. m is the margin value, which is fixed at 1.0 in our experiment.

In the prototype-based domain adaptation total loss $\mathcal{L}_{pro}$, all pairwise relationships between the prototypes of two domains are considered, including intra-class loss $\mathcal{L}_{intra}$ and inter-class loss $\mathcal{L}_{inter}$:

$$\begin{aligned}\mathcal{L}_{pro} = \mathcal{L}_{intra}(D_{\mathcal{S}}, D_{\mathcal{T}}) + \frac{1}{3}(\mathcal{L}_{inter}(D_{\mathcal{S}}, D_{\mathcal{S}}) + \\ \mathcal{L}_{inter}(D_{\mathcal{S}}, D_{\mathcal{T}}) + \mathcal{L}_{inter}(D_{\mathcal{T}}, D_{\mathcal{T}}))\end{aligned} \tag{15}$$

3.3. Image-Level Domain Alignment

In the previous subsection, we introduced prototype-level domain alignment after information aggregation through the GCN network. In this subsection, we will refer to the DA Faster R-CNN [20] idea to align feature representation distributions on the image-level.

In the Faster R-CNN model, image-level representation refers to the output of the feature map after the images pass through the basic convolution layers. In order to eliminate the mismatch of domain distribution on the image-level, we use the patch-based domain classifier as shown in Figure 1.

Specifically, we use each activation from the feature map extracted by the feature extractor to train our domain classifier. Each receptive field of activation corresponding to the input image x_i is a patch, so the domain classifier predicts the domain label on the patch of each image. Image-level domain classification network is used to eliminate image-level domain shift such as image style and illumination. Using the cross entropy loss, the image-level adaptation loss function can be expressed as

$$\mathcal{L}_{img} = -\sum_{i,u,v}\Big[D_i \log p_i^{(u,v)} + (1D_i)\log(1 - p_i^{(u,v)})\Big], \tag{16}$$

where D_i represents the real domain label of the i-th image. When D_i equals 0, the image comes from the source domain. When D_i equals 1, the image is from the target domain. As shown in Figure 3, the domain classifier consists of three convolution layers, and the last one followed by a Softmax layer outputs a probability feature map of two channels. $p_i^{(u,v)}$ represents the predicted value of (u, v) position on the feature map of the i-th image extracted by the feature extraction network of the image-level domain classification network.

In order to align the domain distribution, it is necessary to optimize the parameters of the domain classifier to minimize the domain classification loss and optimize the parameters of the feature extraction network to maximize the domain classification loss at the same time for adversarial training. This operation is implemented by GRL [32], and the gradient descent is used to train the domain classifier. When the gradient is transmitted back to the basic network, the gradient is reversed.

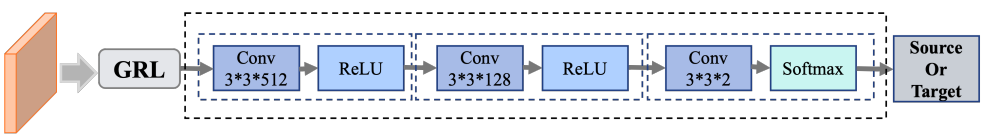


Figure 3. The network structure of image-level domain classifier.

3.4. Overall Training Objective

Detection Network. We choose the famous and powerful Faster R-CNN model as our basic detector. Faster R-CNN is a two-stage detector, mainly divided into three parts: (1) feature extractor, which uses convolutional neural network to extract image features, (2) region proposal network (RPN), responsible for providing candidate regions RoIs, and (3) a region of interest (RoI) header is responsible for classification and regression fine-tuning of RoI generated by RPN. The total loss function of Faster R-CNN is defined as:

$$\mathcal{L}_{det} = \mathcal{L}_{rpn} + \mathcal{L}_{\text{reg}} + \mathcal{L}_{\text{cls}} \tag{17}$$

where $\mathcal{L}_{rpn}$, $\mathcal{L}_{\text{reg}}$, and $\mathcal{L}_{\text{cls}}$ are the loss functions for the RPN, RoI-based regressor, and classifier, respectively.

Overall Training Objective. In our proposed network framework, the optimized loss functions during training include detection network loss $\mathcal{L}_{det}$, rotation-invariant regularizer losses ${\mathcal{L}_{rot}}^{\mathcal{S}}$ and ${\mathcal{L}_{rot}}^{\mathcal{T}}$, image-level feature alignment domain loss $\mathcal{L}_{img}$ and prototype-level alignment $\mathcal{L}_{pro}$. We can write as below:

$$\mathcal{L}_{total} = \mathcal{L}_{det} + \frac{\alpha}{2}\left(\mathcal{L}_{rot}^{\mathcal{S}} + \mathcal{L}_{rot}^{\mathcal{T}}\right) + \beta\mathcal{L}_{img} + \gamma\mathcal{L}_{pro} \tag{18}$$

where α, β, and λ are trade-off parameters to balance the Faster R-CNN detection loss and our newly added domain adaptation components. The network can be trained in an end-to-end manner using a standard SGD algorithm. The overall network in Figure 1 is used in the training phase. During inference, one can remove the domain adaptation components and rotation-invariant component, and simply use the original Faster R-CNN architecture with adapted weights.

4. Experiments and Results

In this section, we first describe the remote sensing image datasets we used, then introduce the evaluation metrics, then introduce the experimental setup, and finally give the quantitative and qualitative results on the three domain adaptation scenarios.

4.1. Datasets

In computer vision, many datasets have been used to evaluate unsupervised domain adaptation for object detection task. In the field of remote sensing, although the task of object detection is also widely studied, such as rotation-invariant object detection [33,34], oriented object detection [38–41], multi-scale object detection [6,42,43], densely packed object detection [44,45] and weakly supervised object detection [46,47], there are relatively little research on cross-domain object detection tasks. In the field of remote sensing, there have been many datasets used for object detection in high-resolution remote sensing images, such as NWPU VHR-10 [48], DOTA [49] and DIOR [50], etc., but few datasets are suitable for unsupervised CDAOD tasks Therefore, we use the WHU change detection dataset [51] and the Inria aerial image dataset [52] with reference to the experimental settings of [27] in our experiment. These two datasets and their preprocessing are described in detail below.

WHU change detection dataset [51]. The WHU change detection dataset contains RGB images covering 20.5 square kilometers in Christchurch, New Zealand in 2012 and 2016, respectively. Christchurch, New Zealand was hit by a 6.3 magnitude earthquake in February 2011. After the earthquake, many buildings were rebuilt or newly built. The aerial image obtained in March 2012 contains 12,796 buildings, and the aerial image in 2016 contains 16,077 buildings. The resolution of these two images is 0.2 m, and the image pixels are 32,507 $\times$ 15,354 as shown in Figure 4. Note that, since the WHU dataset was not built specifically for the object detection task, we use the tightest rectangle in the instance mask as the real bounding box.

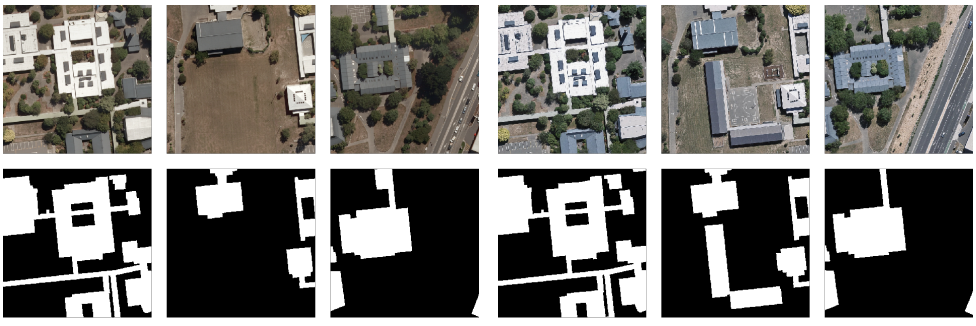
(**a**) 2012_train (**b**) 2016_train (**c**) change label

Figure 4. The WHU change detection dataset. (**a**) is the whole training image of 2012 of the WHU change detection dataset, which is used as the source domain in our experiment; (**b**) is the whole training image of 2016 of the WHU change detection dataset, which is used as the target domain in our experiment; (**c**) is the ground-truth for the change in the WHU change detection dataset.

We cropped the WHU change detection dataset to a 256 × 256 pixel size RGB image. After processing, the samples of the WHU dataset and its label as shown in Figure 5. The image of 2012 is treated as the source domain and 2016 the target domain in our experiment. After the WHU dataset was processed, there were 3839 images from 2012, of which 2598 were used for training and 1341 for testing. For 2016, there are a total of 4200 images, including 2498 from the training set and 1702 from the test set. The details and split of the datasets are shown in Table 1. In our unsupervised domain adaptation task, we did not use the label of the target domain in the training stage.

Table 1. The split of the WHU dataset and the Inria dataset.

Dataset	Domain	Size (Pixels)	Train	Test	Sum
WHU dataset	2012 (source)	256 × 256	2498	1341	3839
	2016 (target)	256 × 256	2498	1702	4200
Inria dataset	Chicago (source)	256 × 256	3023	2391	5414
	Austin (target)	256 × 256	3023	2475	5498



(**a**) 2012_samples (**b**) 2016_samples

Figure 5. The samples of the WHU dataset and its corresponding label. (**a**) are samples of 2012; (**b**) are samples of 2016.

Inria Aerial Image Labeling Dataset [52]**.** The Inria dataset is an aerial image dataset of residential facades in different cities. The dataset includes cities in Europe and the United States with varying residential densities, covering 810 square kilometers. Using the Inria dataset, we expect to be able to account for the in-class variability that occurs over a large geographic range. The dataset has two semantic categories, *building* and *not building*. The image resolution of the dataset is 30 cm and includes RGB bands. In this dataset, only the training set is labeled, and the training set contains five cities: Austin (30°N, 97°W), Chicago (41°N, 87°W), Vienna (48°N, 16°E), Kitsap County (47°N, 22°W), Western Tyrol (47°N, 11°E), each of which contains 36 images of 5000 × 5000 pixels.

In the experimental setting, we selected 24 images from 36 images of Austin and Chicago, two cities with dense buildings. The Chicago image was taken as the source domain and Austin image was taken as the target domain. The preprocessed images are cropped to a 256 × 256 pixel. Chicago has a total of 5414 images, with 3023 in the training set and 2391 in the test set. Austin had a total of 5498 images, 3023 training images, and 2475 test images. Sample examples of the Inria datasets in Chicago and Austin are shown in Figure 6. Similarly, for the Inria dataset, its own labeling is pixel level, so we use the knowledge of connected domain to transform the mask labeling into the bounding box labeling.

(**a**) Chicago (source domain) (**b**) Austin (target domain)

Figure 6. Samples of the Inria Aerial Image Labeling Dataset. Chicago image (**a**) was taken as the source domain and Austin image; (**b**) was taken as the target domain.

4.2. Evaluation Metrics

Generally, the average precision (AP) introduced in VOC2007 [9] is used to evaluate the object detector. The calculation method of precision and recall is as follows:

$$\begin{aligned} \text{Precision} &= \frac{\text{TP}}{\text{TP} + \text{FP}} \\ \text{Recall} &= \frac{\text{TP}}{\text{TP} + \text{FN}} \end{aligned} \tag{19}$$

Precision is actually the ratio of true positives (positive samples are correctly identified as positive samples) in the identified images. Recall is the proportion of all positive samples in the test set that are correctly identified as positive samples.

To determine whether the detector's prediction matches the ground-truth, an intersection over union (IoU) metric is used. The definition of this measure is as follows:

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{20}$$

The IoU metric is used to calculate whether a particular prediction is a true positive or a false positive. When it is greater than the set threshold, it is judged as a true positive; otherwise, it is judged as a false positive.

Furthermore, in order to compare the overall performance of all categories, the category-wise mean average precision (mAP) index is used, and the calculation formula is as follows:

$$\text{mAP} = \frac{\sum_{i=1}^{C} \text{AP}_c}{C}, \tag{21}$$

where C is the total number of category. For our building detection datasets, category C is equal to 1 so that the mAP is equal to the AP of house class.

4.3. Implementation Details

Training details. In our experiment, we use ResNet-50 [3] as the basic network structure and Faster R-CNN as the frame network. Specifically, the models are initialized using weights pretrained on the ImageNet [53]. The rotation-invariant regularizer is embedded on the RoI pooling layer with 2048 neurons. This work is implemented by Pytorch [54] deep learning framework and trained on a single Nvidia GeForce RTX 2080Ti GPU with 12 GB RAM. As an optimizer for the training, we used a stochastic gradient descent (SGD) optimizer [55] with the initial learning rate set to 0.001 and momentum is set to 0.9 and weight decay is 5×10^{-4}. In our implementation, the batch size is set to 4, two images for the source domain and two images for the target domain. The number of total training epoch is set as 20, and the learning rate warm-up strategy [3] is used in the first 200 iterations of training. In the training process, for each image, 128 anchors are sampled with a positive ratio of 0.25. For evaluation, we report mAP with an IoU threshold of 0.5. Our code is modified based on DA-Faster's [20] code (https://github.com/yuhuayc/da-faster-rcnn (accessed on 25 August 2021)).

Parameter Settings. In the rotation-invariant regularizer module, we set rotation transformations $T_\phi = \{T_{\phi_1}, T_{\phi_2}, \cdots, T_{\phi_K}\}$ with $\phi = \{45°, 90°, 150°\}$ to perform the rotation operation on the training images in the source domain and the target domain to obtain $4 \times N$ samples. The bounding box visualization of the image before and after rotation is shown in Figure 7 below. The missing values generated by the rotation of 45° and 150° were filled with background.

(**a**) Original image (**b**) 0° rotation (**c**) 45° rotation (**d**) 90° rotation (**e**) 150° rotation

Figure 7. Visualization of bounding boxes of the original and rotated images. (**a**) is the original image obtained from the Inria dataset; (**b**) is the visualization of the original image's bounding box; (**c**–**e**) are, respectively, the visualization of the rotated image's bounding box.

4.4. Results

Scenario settings. To fully verify the effectiveness of cross-domain remote sensing image object detection, we conduct experiments by using WHU and Inria datasets. The source training images have their annotations and target training data images are unlabeled. In detail, we provide three different domain shift scenarios experimental settings:

1. WHU 2012 → WHU 2016, same data source and region, different times;

2. Inria (Chicago) → Inria (Austin), same data source, different regions;
3. WHU 2012 → Inria (Austin), different data source, different regions.

Performance comparison. To validate the effectiveness of the proposed approach, we compared our method with six baselines: Faster R-CNN model trained only on source examples without any domain adaptation (Source-only), Faster R-CNN model with a CycleGAN transfer adaptation (CycleGAN) [56], Faster R-CNN model with an AgGAN transfer adaptation (AgGAN) [57], domain adaptation Faster R-CNN based on adversarial training at the image level (DA-Faster*) [20], domain adaptation Faster R-CNN based on adversarial training at the image and instance level (DA-Faster) [20], and Faster R-CNN model trained on the labeled target samples (Oracle).

Ablation study. To further verify the effects of the rotation-invariant regularizer module and the prototype-Level alignment based on the relation-aware graph on the entire model, we conducted ablation studies. Among them, "img-level" represents the domain alignment at the image-level, "ins-level" represents the domain alignment at the instance-level, "pro-level" represents the domain alignment at the prototype-level, and "rot-inv" represents the rotation-invariant regularizer module.

4.4.1. Cross-Time. WHU 2012 → WHU 2016

In this experiment, the WHU 2012 dataset and the WHU 2016 dataset serve as the source domain and target domain, respectively. The results are reported in the test set of the WHU 2016 dataset. It aims to perform adaptation across different times in the same region, which is Christchurch, New Zealand. In this scenery, although the source area and the target area are the same areas, with the development of social economy and urban expansion, the buildings of the city have obvious changes.

In Table 2, the comparison between our method and other CDAOD methods is presented on the building category. It can be seen from Table 2 that our method outperforms all the methods. Compared with DA-Faster and DA-Faster*, our method is 2.09% and 3.64% higher, respectively. Compared with GAN-based methods, our method is 6.98% higher than CycleGAN and 4.44% higher than AgGAN. Compared with the method without domain adaptation (Source-only), our method achieves an AP of 77.09% and a gain of +5.1%. The domain adaptation method DA-Faster performs well in the field of computer vision, but it does not perform well when used directly on remote sensing images, which is related to the complex structure of remote sensing images. It can be noted that this cross-time urban building detection problem is actually a change detection problem. The source domain and target domain have many similarities, so the overall detection accuracy is relatively high.

Table 2. Mean average precision (%) of cross-time adaptation scenario, WHU 2012 → WHU 2016.

Method	Img-Level	Ins-Level	Pro-Level	Rot-Inv	House AP
Source-only					0.7199
CycleGAN	√(GAN-based)				0.7011
AgGAN	√(GAN-based)				0.7265
DA-Faster*	√				0.7345
DA-Faster	√	√			0.7500
Our(pro)	√		√		0.7628
Our(inv)	√			√	0.7599
Our(pro+inv)	√		√	√	**0.7709**
Oracle					0.8143

It can be seen from the ablation experiments in Table 2 that the rotation-invariant regularizer module (rot-inv) and the prototype-level alignment module (pro-level) in the network we designed can significantly improve the domain adaptation performance of the model, indicating that our model can well solve the task of CDAOD in remote sensing

images. We can also know that the prototype-level alignment module is better than the rotation-invariant regularizer module, which is 4.49% and 4% higher than the Source-only method, respectively.

Figure 8 displays some typical qualitative detection results on the task WHU 2012 → WHU 2016, in which Source-only, CycleGAN, AgGAN, DA-Faster, and our approach are evaluated. As shown in the figure, the Source-only model can poorly localize objects. DA-Faster predicts bounding box more accurately, but it incorrectly classifies the road as a building, and produces some false positives. Our model successfully inhibits false positives, and it is able to localize objects precisely even when severe occlusion occurs.

(**a**) Source-only (**b**) CycleGAN (**c**) AgGAN (**d**) DA-Faster (**e**) Ours (**f**) Oracle (**g**) GT

Figure 8. Visualization of detection results on the task WHU 2012 → WHU 2016.

4.4.2. Cross-Region. Inria (Chicago) → Inria (Austin)

Different urban remote sensing data, although the data collection method is the same, due to the characteristics of the city itself will produce domain shift. To adopt a building detection model from one city to another, during the training phase, we utilize the training set of Chicago in Inria as the source domain and Austin in Inria as the target domain. We report the performance in the test set of the Austin in Inria.

As shown in Table 3, we achieve an mAP of 51.54% on the cross-region adaptation scenario task, which is the best result among all the approaches. In particular, we achieve a remarkable increase of +14.14% gain over the Source-only. Other methods CycleGAN, AgGAN, DA-Faster*, and DA-Faster correspond to Source-only gains of −4.45%, −2.15%, 0.39%, and 12.06%, respectively. It can be concluded that the method based on GAN does not perform well in cross-domain object detection in remote sensing images. In the ablation experiment, our mAP improves by +13.23% compared with Source-only if only using the prototype-level alignment module (pro) and by +12.69% compared with Source-only if only adding the rotation-invariant regularizer module (inv). It can be seen from the visualized results in Figure 9 that the performance of our proposed method is greatly improved compared to the baseline methods, and many false detections and missed detections have been improved.

Table 3. Mean average precision (%) of cross-region adaptation scenario, Inria (Chicago) → Inria (Austin).

Method	Img-Level	Ins-Level	Pro-Level	Rot-Inv	House AP
Source-only					0.3740
CycleGAN	√(GAN-based)				0.3465
AgGAN	√(GAN-based)				0.3525
DA-Faster*	√				0.3779
DA-Faster	√	√			0.4946
Our(pro)	√		√		0.5063
Our(inv)	√			√	0.5009
Our(pro+inv)	√		√	√	0.5154
Oracle					0.7061

(**a**) Source-only (**b**) CycleGAN (**c**) AgGAN (**d**) DA-Faster (**e**) Ours (**f**) Oracle (**g**) GT

Figure 9. Visualization of detection results on the task Inria (Chicago) → Inria (Austin).

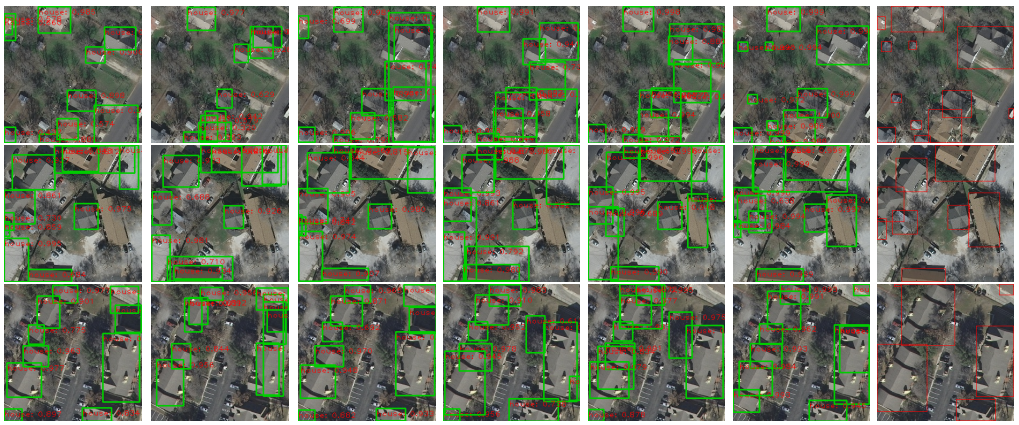
4.4.3. Cross Data Source and Cross-Region. WHU 2012 → Inria (Austin)

Different places, different resolutions, and different data acquisition methods will seriously incur the difference of ground objects in remote sensing images, resulting in domain shift problem. To simulate this adaptation, in the training phase, we use the training set of WHU 2012 dataset and Austin in Inria as the source and target domain, respectively. The results are reported in the testing set of the Austin in Inria.

Table 4 displays the results on the cross data source and cross-region adaptation task. As shown in the table, we achieved 47.29% of the mAP on this domain adaptation task, achieving the best results. Specifically, compared with the Source-only model, we achieved a significant gain increase of +18.15%. Among all GAN-based domain adaptation methods, our mAP is +26.13% better than CycleGAN and +16.23% better than AgGAN. Compared with the domain adaptation methods DA-Faster* and DA-Faster based on adversarial training, our method improves 13.51% and 2.4%, respectively. This result implies that the domain alignment performance at the image-level is not good. In addition, we can also see in the visualization results in Figure 10 that our method has achieved the best results. It is worth noting that comparing scenario 2 (Inria (Chicago) → Inria (Austin)) and scenario 3 (WHU 2012 → Inria (Austin)), the AP of Table 4 is not as high as that shown in Table 3 due to the huge discrepancy between the domains of scenario 3. These results show that our method can effectively improve the domain shift caused by different data acquisition methods, resolution, scale, regional differences, and other problems. From the ablation experiment in Table 4, it can be seen that the prototype-level alignment (pro) and rotation-invariant regularizer (inv) modules can effectively improve the overall performance of cross-domain object detection, and have an AP improvement of 13.23% and 12.69%, respectively, compared to Source-only.

Table 4. Mean average precision (%) of cross data source and cross-region adaptation scenario, WHU 2012 → Inria (Austin).

Method	Img-Level	Ins-Level	Pro-Level	Rot-Inv	House AP
Source-only					0.2914
CycleGAN	√(GAN-based)				0.2116
AgGAN	√(GAN-based)				0.3106
DA-Faster*	√				0.3378
DA-Faster	√	√			0.4478
Our(pro)	√		√		0.4489
Our(inv)	√			√	0.4486
Our(pro+inv)	√		√	√	0.4729
Oracle					0.7061



(**a**) Source-only (**b**) CycleGAN (**c**) AgGAN (**d**) DA-Faster (**e**) Ours (**f**) Oracle (**g**) GT

Figure 10. Visualization of detection results on the task Inria WHU 2012 → Inria (Austin).

5. Discussion

In Section 4.4, we have shown that the results of our proposed method are superior to other domain adaptation methods, and can well handle the domain discrepancy of remote sensing images. In this section, we will analyze the source of the error and do a sensitivity analysis of the parameters.

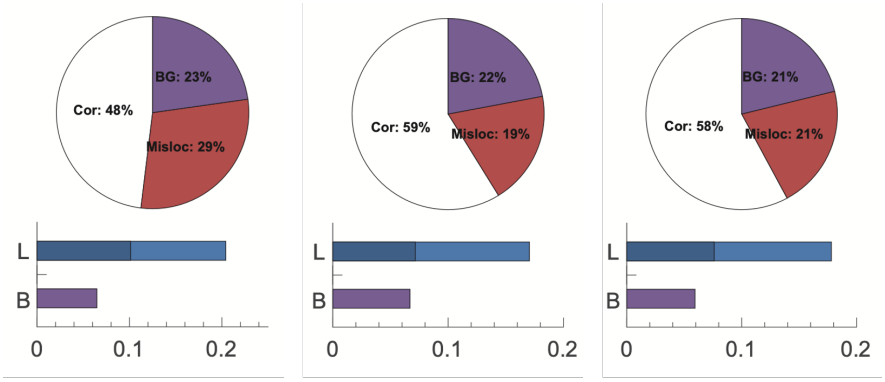
5.1. Performance Analysis Focusing on Errors

In the previous sections, we have shown that both prototype-level alignment and rotation-invariant regularizer help reduce domain discrepancy. To further validate the individual effect of prototype-level adaptation and rotation-invariant feature, we analyze the detection errors for models using different components on different levels.

We study **Inria (Chicago) → Inria (Austin)** case for the analysis. Since the Austin test contains 2475 images, we select top-N predictions with highest confidence for the vanilla Faster R-CNN model, our model with only prototype-level adaptation, and our model with only rotation-invariant regularizer, respectively, where N is the number of the objects in category. Inspired by [58], we categorize the detections into three error types: **correct:** IoU with ground-truth ≥ 0.5. **mis-localized:** $0.3 \leq$ IoU with ground-truth < 0.5, and **background:** IoU with ground-truth < 0.3.

The results are shown in Figure 11. From the figure, we can observe that each component (prototype-level alignment or rotation-invariant regularizer) increases the number of correct detections (white color), and significantly reduces the number of false positives (other colors). In addition, we have also observed that the model using rotation-invariant regularizer (Rot. Only) produces higher mislocalization errors than the model using prototype-level alignment (Pro. Only). The reason may be that prototype-level alignment

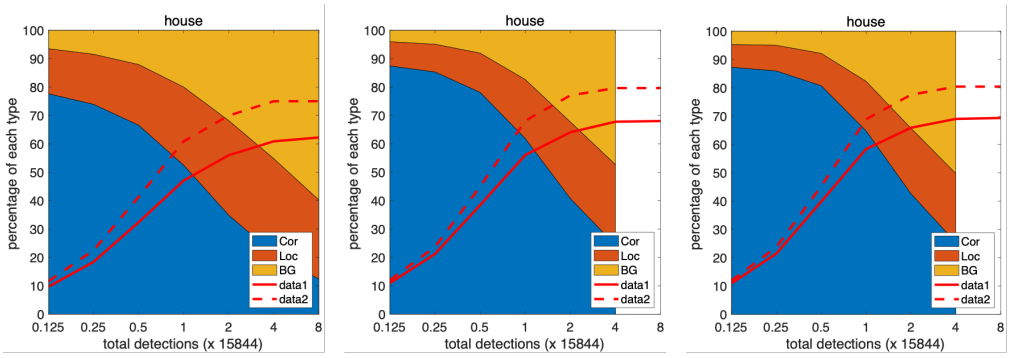
aggregates features captured by RoI more directly, resulting in better alignment. As you can see from the bar chart below, our approach reduces location errors and background errors and turns false positives into true positives.



(**a**) Faster R-CNN (**b**) Ours (Pro. Only) (**c**) Ours (Rot. Only)

Figure 11. Analysis of Top-Ranked Detections on **Inria (Chicago) → Inria (Austin)**. Pie chart: The scores of the top N detections that are correct (Cor) or false positives due to poor positioning (Misloc) or confusion with background or unlabeled objects (BG). Bar graph: Absolute AP improvement obtained by eliminating all false positives of one type. 'B': Will not be confused with background objects. 'L': If the bad positioning is removed, the first bar is an improvement; if the positioning error is corrected and the false positive becomes a true positive, the second column is an improvement.

We continue the error analysis of the adaptation case on **Inria (Chicago) → Inria (Austin)**. Figure 12 shows the error analysis in the Austin test set. Comparing the baseline Source-only with the DA-Faster, we observe that domain adaptation improves the detection performance. Comparing DA-Faster and our proposed method, we observe that our method slightly reduces localization errors and background errors.



(**a**) Source-only (**b**) DA-Faster (**c**) Ours

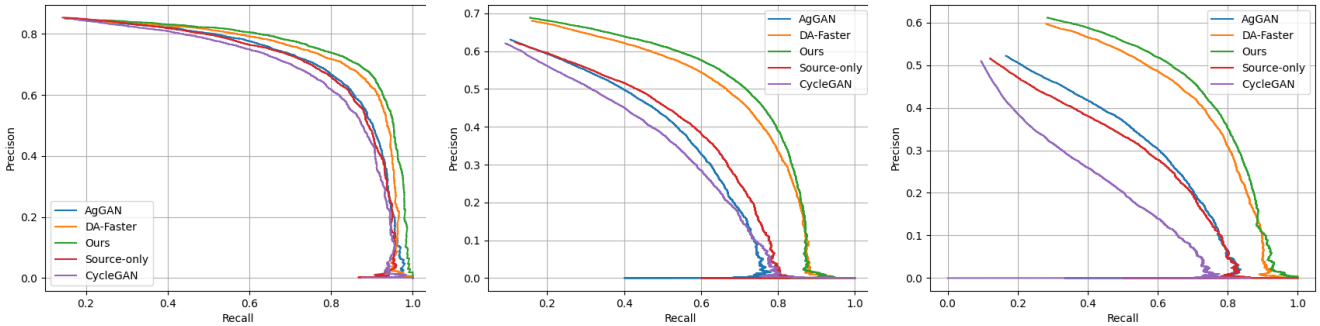
Figure 12. Visualization of performance for various methods on **Inria (Chicago) → Inria (Austin).** The red solid line and the red dashed line respectively reflect the changes in the recall of the strong standard (0.5 IoU) and the weak standard (0.1 IoU) as the number of inspections increases.

5.2. Analysis of P–R curve

In computer vision, precision and recall are two important performance indicators, and we can use the P–R curve [59] to see the impact of these two factors on model performance.

From the P–R curve in Figure 13, it can be intuitively seen that our method (the outermost green curve in the figure) achieved the best results on the three domain adaptation scenarios, while the GAN-based method (i.e., AgGAN and CycleGAN) did not have good results. The DA-faster method with domain alignment from image-level and instance-level union is superior to the method without domain adaptation (i.e., Source-only). It can be

seen intuitively from the P–R curve in Figure 13 that our method (the outermost green curve in the figure) has achieved the best results in the three domain adaptation scenarios, with overall higher precision and recall than other methods. The next best method is the DA-Faster method (orange curve), which is significantly better than Source-only (red curve) without domain adaptation. However, GAN-based methods CycleGAN (purple curve) and AgGAN (blue curve) performed poorly in domain adaptation on image style appearance for our three cross-domain tasks.



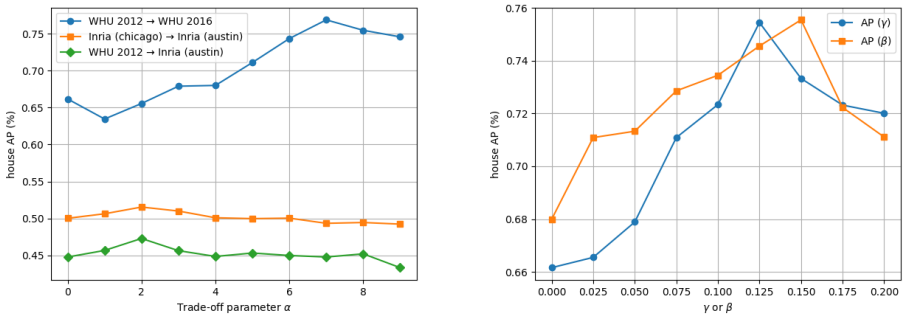
(**a**) WHU 2012 → WHU 2016 (**b**) Inria (Chicago) → Inria (Austin) (**c**) WHU 2012 → Inria (Austin)

Figure 13. P–R curves of three different domain adaptation scenarios. The different colored curves in the figure represent different cross-domain adaptation methods.

5.3. Parameter Sensitivity

In this experiment, we verified the sensitivity of our method to α, β, and γ, which trades off between detection and domain adaptation loss. First of all, we discuss the choice of the parameter α, which is used to balance the rotation-invariant loss with other losses. Figure 14a shows the result of our method for parameter α when the other parameter β and γ is fixed, and all results are evaluated on the three domain adaptation scenario. As can be seen from the line chart, it is better to set the parameter α to 7 for the WHU 2012 → WHU 2016 task and 2 for the other two tasks. For task Inria (Chicago) → Inria (Austin) and task WHU 2012 → Inria (Austin), the change of parameter α from 1 to 10 does not have a great impact on the detection effect. This may be because for these two tasks, the domain discrepancy is larger than that of task WHU 2012 → WHU 2016.

Figure 14b shows results for parameter sensitivity of β and γ in Equation (18), and we evaluate the AP value of the final building detection result. We conduct experiments on adaptation task: Inria (Chicago) → Inria (Austin), where parameter α is set to 2, and we evaluate the AP value of the building detection result. We achieve the best performance when $\beta = 0.15$ and $\gamma = 0.125$ and the best accuracy is 75.78%. Under this parameter setting, the image-level and relation-aware prototype-level alignment benefits domain adaptation most.



(**a**) The sensitivity of parameter α. (**b**) The sensitivity of parameter β and γ.

Figure 14. Parameter Sensitivity on α, β, and γ.

6. Conclusions

It is risky to apply the object detector trained in one remote sensing scenario directly to a new one because the gap between the two domains can seriously degrade the performance of the model. In this paper, we propose a Rotation-Invariant and Relation-Aware CDAOD network for optical remote sensing images. Our network uses a rotation-invariant regularizer term to solve the problem of rotation diversity of remote sensing images, and constructs a relation-aware graph to obtain the class-based prototype representation, so as to obtain alignment at the prototype-level. In addition, we have added image-level domain classifiers for feature alignment. We use the WHU change detection dataset and the Inria aerial image dataset to demonstrate the effectiveness of our proposed method. Our approach achieves optimal performance in all three cross-domain scenarios compared to other competitive approaches. It shows that our method can adapt well to remote sensing scenes with obvious domain discrepancy and provide an effective and feasible solution for cross-domain object detection in remote sensing images.

Author Contributions: Conceptualization, Y.C., and X.M.; methodology, Y.C.; validation, Y.C., X.M., and Q.L.; formal analysis, X.M.; resources, T.W., B.W., and X.M.; data curation, Y.C., and X.M.; writing—original draft preparation, Y.C., and Q.L.; writing—review and editing, Y.C., X.M., and Q.L.; visualization, Y.C., and Q.L.; supervision, T.W., B.W., and X.M.; project administration, Y.C., and X.M.; funding acquisition, T.W., B.W., and X.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (NSFC) under Grant No. 41971352, National Key Research and Development Program of China under Grant No. 2018YFB0505003.

Data Availability Statement: Not applicable.

Acknowledgments: The authors are very grateful to the many people who helped to comment on the article, and the Large Scale Environment Remote Sensing Platform (Facility No. 16000009, 16000011, 16000012) provided by Wuhan University. Special thanks to editors and reviewers for providing valuable insight into this article.

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

References

1. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
2. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
3. He, K.; Zhang, X.; Ren, S.; Sun, J. [CrossRef] Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef]
4. Dai, J.; Li, Y.; He, K.; Sun, J. R-FCN: Object Detection via Region-based Fully Convolutional Networks. *arXiv* **2016**, arXiv:abs/1605.06409.
5. Redmon, J.; Divvala, S.; Girshick, R.B.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
6. Lin, T.Y.; Dollár, P.; Girshick, R.B.; He, K.; Hariharan, B.; Belongie, S.J. Feature Pyramid Networks for Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 June 2017; pp. 936–944.
7. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
8. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.E.; Fu, C.Y.; Berg, A. SSD: Single Shot MultiBox Detector. In *European Conference on Computer Vision (ECCV 2016)*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
9. Everingham, M.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
10. Lin, T.Y.; Maire, M.; Belongie, S.J.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. In *European Conference on Computer Vision (ECCV 2014)*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 740–755.

11. Shrivastava, A.; Shekhar, S.; Patel, V. 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.
12. 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*, 59:1–59:35.
13. Saito, K.; Ushiku, Y.; Harada, T. Asymmetric Tri-training for Unsupervised Domain Adaptation. In Proceedings of the 34th International Conference on Machine Learning, Sydney, NSW, Australia, 6–11 August 2017.
14. Kurmi, V.; Kumar, S.; Namboodiri, V.P. Attending to Discriminative Certainty for Domain Adaptation. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–19 June 2019; pp. 491–500.
15. Hoffman, J.; Wang, D.; Yu, F.; Darrell, T. FCNs in the Wild: Pixel-level Adversarial and Constraint-based Adaptation. *arXiv* **2016**, arXiv:abs/1612.02649.
16. Tsai, Y.H.; Hung, W.; Schulter, S.; Sohn, K.; Yang, M.H.; Chandraker, M. Learning to Adapt Structured Output Space for Semantic Segmentation. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7472–7481.
17. Zou, Y.; Yu, Z.; Kumar, B.; Wang, J. Unsupervised Domain Adaptation for Semantic Segmentation via Class-Balanced Self-training. In *European Conference on Computer Vision (ECCV 2018)*; Springer: Berlin/Heidelberg, Germany, 2018.
18. Tsai, Y.H.; Sohn, K.; Schulter, S.; Chandraker, M. Domain Adaptation for Structured Output via Discriminative Patch Representations. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–3 November 2019; pp. 1456–1465.
19. Li, G.; Kang, G.; Liu, W.; Wei, Y.; Yang, Y. Content-Consistent Matching for Domain Adaptive Semantic Segmentation. In *European Conference on Computer Vision (ECCV 2020)*; Springer: Berlin/Heidelberg, Germany, 2020.
20. Chen, Y.; Li, W.; Sakaridis, C.; Dai, D.; Gool, L. Domain Adaptive Faster R-CNN for Object Detection in the Wild. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 3339–3348.
21. Saito, K.; Ushiku, Y.; Harada, T.; Saenko, K. Strong-Weak Distribution Alignment for Adaptive Object Detection. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–19 June 2019; pp. 6949–6958.
22. Khodabandeh, M.; Vahdat, A.; Ranjbar, M.; Macready, W. A Robust Learning Approach to Domain Adaptive Object Detection. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–3 November 2019; pp. 480–490.
23. Kim, S.; Choi, J.; Kim, T.; Kim, C. Self-Training and Adversarial Background Regularization for Unsupervised Domain Adaptive One-Stage Object Detection. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–3 November 2019; pp. 6091–6100.
24. Cai, Q.; Pan, Y.; Ngo, C.; Tian, X.; Duan, L.; Yao, T. Exploring Object Relation in Mean Teacher for Cross-Domain Detection. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–19 June 2019; pp. 11449–11458.
25. Xu, M.; Wang, H.; Ni, B.; Tian, Q.; Zhang, W. Cross-Domain Detection via Graph-Induced Prototype Alignment. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Virtual, 15–18 June 2020; pp. 12352–12361.
26. Deng, J.; Li, W.; Chen, Y.; Duan, L. Unbiased Mean Teacher for Cross Domain Object Detection. *arXiv* **2020**, arXiv:abs/2003.00707.
27. Li, X.; Luo, M.; Ji, S.; Zhang, L.; Lu, M. Evaluating generative adversarial networks based image-level domain transfer for multi-source remote sensing image segmentation and object detection. *Int. J. Remote Sens.* **2020**, *41*, 7343 – 7367. [CrossRef]
28. Koga, Y.; Miyazaki, H.; Shibasaki, R. A Method for Vehicle Detection in High-Resolution Satellite Images that Uses a Region-Based Object Detector and Unsupervised Domain Adaptation. *Remote Sens.* **2020**, *12*, 575. [CrossRef]
29. Ding, J.; Xue, N.; Xia, G.; Bai, X.; Yang, W.; Yang, M.Y.; Belongie, S.J.; Luo, J.; Datcu, M.; Pelillo, M.; et al. Object Detection in Aerial Images: A Large-Scale Benchmark and Challenges. *arXiv* **2021**, arXiv:abs/2102.12219.
30. Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; Schiele, B. The Cityscapes Dataset for Semantic Urban Scene Understanding. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 3213–3223.
31. Johnson-Roberson, M.; Barto, C.; Mehta, R.; Sridhar, S.N.; Rosaen, K.; Vasudevan, R. Driving in the Matrix: Can virtual worlds replace human-generated annotations for real world tasks? In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 746–753.
32. Ganin, Y.; Lempitsky, V. Unsupervised Domain Adaptation by Backpropagation. *arXiv* **2015**, arXiv:abs/1409.7495.
33. Cheng, G.; Han, J.; Zhou, P.; Xu, D. Learning Rotation-Invariant and Fisher Discriminative Convolutional Neural Networks for Object Detection. *IEEE Trans. Image Process.* **2019**, *28*, 265–278. [CrossRef]
34. Cheng, G.; Zhou, P.; Han, J. RIFD-CNN: Rotation-Invariant and Fisher Discriminative Convolutional Neural Networks for Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2884–2893.

35. Girshick, R.B. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Washington, DC, USA, 7–13 December 2015; pp. 1440–1448.
36. He, C.H.; Lai, S.C.; Lam, K.M. Improving Object Detection with Relation Graph Inference. In Proceedings of the 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 2537–2541. [CrossRef]
37. Kipf, T.; Welling, M. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv* **2017**, arXiv:abs/1609.02907.
38. Ding, J.; Xue, N.; Long, Y.; Xia, G.; Lu, Q. Learning RoI Transformer for Oriented Object Detection in Aerial Images. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–19 June 2019; pp. 2844–2853.
39. Xu, Y.; Fu, M.; Wang, Q.; Wang, Y.; Chen, K.; Xia, G.; Bai, X. Gliding Vertex on the Horizontal Bounding Box for Multi-Oriented Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 1452–1459. [CrossRef]
40. Yang, X.; Liu, Q.; Yan, J.; Li, A. R3Det: Refined Single-Stage Detector with Feature Refinement for Rotating Object. In Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI-21), Virtual, 2–9 February 2021.
41. Li, C.; Xu, C.; Cui, Z.; Wang, D.; Jie, Z.; Zhang, T.; Yang, J. Learning Object-Wise Semantic Representation for Detection in Remote Sensing Imagery. In Proceedings of the Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019.
42. Cheng, G.; Si, Y.; Hong, H.; Yao, X.; Guo, L. Cross-Scale Feature Fusion for Object Detection in Optical Remote Sensing Images. *IEEE Geosci. Remote Sens. Lett.* **2021**, *18*, 431–435. [CrossRef]
43. Qian, X.; Lin, S.; Cheng, G.; Yao, X.; Ren, H.; Wang, W. Object Detection in Remote Sensing Images Based on Improved Bounding Box Regression and Multi-Level Features Fusion. *Remote Sens.* **2020**, *12*, 143. [CrossRef]
44. Bodla, N.; Singh, B.; Chellappa, R.; Davis, L. Soft-NMS — Improving Object Detection with One Line of Code. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 5562–5570.
45. Pan, X.; Ren, Y.; Sheng, K.; Dong, W.; Yuan, H.; Guo, X.W.; Ma, C.; Xu, C. Dynamic Refinement Network for Oriented and Densely Packed Object Detection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Virtual, 15–18 June 2020; pp. 11204–11213.
46. Feng, X.; Han, J.; Yao, X.; Cheng, G. Progressive Contextual Instance Refinement for Weakly Supervised Object Detection in Remote Sensing Images. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 8002–8012. [CrossRef]
47. Yao, X.; Feng, X.; Han, J.; Cheng, G.; Guo, L. Automatic Weakly Supervised Object Detection From High Spatial Resolution Remote Sensing Images via Dynamic Curriculum Learning. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 675–685. [CrossRef]
48. Cheng, G.; Han, J. A Survey on Object Detection in Optical Remote Sensing Images. *arXiv* **2016**, arXiv:abs/1603.06201.
49. Xia, G.S.; Bai, X.; Ding, J.; Zhu, Z.; Belongie, S.J.; Luo, J.; Datcu, M.; Pelillo, M.; Zhang, L. DOTA: A Large-Scale Dataset for Object Detection in Aerial Images. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 3974–3983.
50. Li, K.; Wan, G.; Cheng, G.; Meng, L.; Han, J. Object Detection in Optical Remote Sensing Images: A Survey and A New Benchmark. *arXiv* **2019**, arXiv:abs/1909.00133.
51. Ji, S.; Wei, S.; Lu, M. Fully Convolutional Networks for Multisource Building Extraction From an Open Aerial and Satellite Imagery Data Set. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 574–586. [CrossRef]
52. Maggiori, E.; Tarabalka, Y.; Charpiat, G.; Alliez, P. Can semantic labeling methods generalize to any city? the inria aerial image labeling benchmark. In Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017; pp. 3226–3229.
53. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Li, F.-F. L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 22–24 June 2009; pp. 248–255.
54. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A. Automatic differentiation in PyTorch. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017.
55. Loshchilov, I.; Hutter, F. SGDR: Stochastic Gradient Descent with Restarts. *arXiv* **2016**, arXiv:abs/1608.03983.
56. Zhu, J.; Park, T.; Isola, P.; Efros, A.A. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. *arXiv* **2017**, arXiv:abs/1703.10593.
57. Tang, H.; Xu, D.; Sebe, N.; Yan, Y. Attention-Guided Generative Adversarial Networks for Unsupervised Image-to-Image Translation. *arXiv* **2019**, arXiv:abs/1903.12296.
58. Hoiem, D.; Chodpathumwan, Y.; Dai, Q. Diagnosing Error in Object Detectors. In Proceedings of the 12th European Conference on Computer Vision, Florence, Italy, 7–13 October 2012.
59. Davis, J.; Goadrich, M. The Relationship between Precision-Recall and ROC Curves. In Proceedings of the 23rd International Conference on Machine Learning; Association for Computing Machinery: New York, NY, USA, 2006; pp. 233–240. [CrossRef]