# Cross-domain Detection via Graph-induced Prototype Alignment

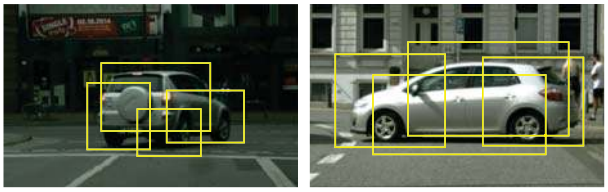
Minghao Xu[1,2] Hang Wang[1,2] Bingbing Ni[1,2,3*] Qi Tian[4] Wenjun Zhang[1]
[1]Shanghai Jiao Tong University, Shanghai 200240, China
[2]MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University
[3]Huawei Hisilicon [4]Huawei Noah's Ark Lab
{xuminghao118, wang–hang, nibingbing, zhangwenjun}@sjtu.edu.cn
nibingbing@hisilicon.com tian.qi1@huawei.com

## Abstract

*Applying the knowledge of an object detector trained on a specific domain directly onto a new domain is risky, as the gap between two domains can severely degrade model's performance. Furthermore, since different instances commonly embody distinct modal information in object detection scenario, the feature alignment of source and target domain is hard to be realized. To mitigate these problems, we propose a Graph-induced Prototype Alignment (GPA) framework to seek for category-level domain alignment via elaborate prototype representations. In the nutshell, more precise instance-level features are obtained through graph-based information propagation among region proposals, and, on such basis, the prototype representation of each class is derived for category-level domain alignment. In addition, in order to alleviate the negative effect of class-imbalance on domain adaptation, we design a Class-reweighted Contrastive Loss to harmonize the adaptation training process. Combining with Faster R-CNN, the proposed framework conducts feature alignment in a two-stage manner. Comprehensive results on various cross-domain detection tasks demonstrate that our approach outperforms existing methods with a remarkable margin. Our code is available at* `https://github.com/ChrisAllenMing/GPA-detection`.

## 1. Introduction

Following the rapid development of techniques leveraging Deep Neural Networks (DNNs), a variety of computer-vision-related tasks, *e.g.* object classification [20, 14], object detection [35, 24], and semantic segmentation [4, 13], witnessed major breakthroughs in the last decade. It should be noticed that the impressive performance of these models is established, to a great extent, on the basis of massive amounts of annotated data, of which the annotation process itself could be a laborious task in many cases. Furthermore, when the model trained on a domain with abundant annotations is applied to a distinct domain with limited, even unavailable, labels, it will suffer from performance decay, due to the existence of domain shift [53].



(a) (b)

Figure 1. Two vehicles and corresponding region proposals from the Cityscapes [6] dataset which serves as target domain. These two vehicles reflect multi-modal information, *e.g.* distinct scale and orientation, and the generated region proposals contain incomplete information of them.

One of the extensively explored techniques to deal with such dilemma is Unsupervised Domain Adaptation (UDA), which seeks for knowledge transfer from a labeled dataset (source domain) to another unlabeled one (target domain). In order to encourage domain-invariant feature representations, commonly adopted strategies can be roughly classified into two categories: 1) Minimizing an explicitly defined domain discrepancy measurement [25, 44, 42, 48]; 2) Applying adversarial training to UDA via domain classifier [7, 43, 1, 32]. These strategies are comprehensively exploited in classification-based tasks.

Besides classification, cross-domain detection is also strongly demanded in modern Computer Vision systems, including intelligent surveillance and autonomous driving, in which the deployment environment, *e.g.* backgrounds, weather, illumination, changes from site to site. Previous works [5, 2, 55] utilize independent or grouped region proposals to align source and target domain on local instance level. However, since supervisory signal is lacked on target domain, the generated region proposals commonly deviate from instances, which makes the information from primal

*The corresponding author is Bingbing Ni.

proposals improper to depict corresponding instances. In addition, the representation of an instance is insufficient to characterize the category it belongs to, because a single instance can only reflect limited modal information, *e.g.* specific scale or orientation. However, the representations of instances within a category are multi-modal. Two typical examples are illustrated in Figure 1, where two vehicles express different modal information, and the generated region proposals deviate from objects. These two problems make instance-level domain alignment trapped into dilemma. Except for these issues, in multi-class cross-domain detection tasks, class-imbalance leads to the inconsistency of domain adaptation process among different classes along training, which greatly impairs model's adaptation performance on those sample-scarce categories.

Motivated by these problems, we propose the **Graph-induced Prototype Alignment** (GPA) framework and embed it into a two-stage detector, Faster R-CNN [35]. For the sake of better local alignment via region proposals, we introduce two key components, *graph-based region aggregation* and *confidence-guided merging*. In graph-based region aggregation, a relation graph which takes both the location and size of proposals into consideration is constructed to aggregate features on instance level, such that the critical features of each instance are integrated. In confidence-guided merging, the multi-modal information contained in various instances is embodied by prototype[†] representations, such that, by utilizing the complementarity of multi-modal information, each category can be better characterized. Using prototypes as the proxy of different classes, category-level domain alignment is performed. Furthermore, considering that class-imbalance exists in the multi-class cross-domain detection tasks, we harmonize the process of domain adaptation via a *Class-reweighted Contrastive Loss*, in which the sample-scarce classes are assigned with higher weights, thus they can be better aligned during training.

Based on the two-stage structure of Faster R-CNN, we also conduct feature alignment in a two-stage manner: 1) In the first stage, foreground and background distributions are separated, and class-agnostic alignment is performed on feature distributions of two domains; 2) In the second stage, more fine-grained alignment is respectively performed on each foreground category.

Our contributions can be summarized as follows:

- We propose the Graph-induced Prototype Alignment (GPA) framework, in which more precise instance-level features are obtained through graph-based region aggregation, and prototype representations are derived for category-level domain alignment.
- In multi-class cross-domain detection tasks, for tackling the class-imbalance during feature alignment, we design a Class-reweighted Contrastive Loss to harmonize the adaptation process among different classes.
- Combining with the Faster R-CNN architecture, we propose a two-stage domain alignment scheme, and it achieves state-of-the-art performance on the cross-domain detection tasks under various scenarios.

[†]Prototype is the representative embedding of all samples within the same class.

## 2. Related Work

**Object Detection.** Current object detection methods can be roughly categorized into two classes: one-stage detectors [33, 24, 34, 22] and two-stage detectors [10, 9, 35, 21, 13]. R-CNN [10] first obtains region proposals with selective search and then classifies each proposal. Fast R-CNN [9] speeds up detection process by introducing RoI pooling. Faster R-CNN [35] produces nearly cost-free region proposals with Region Proposal Network. One-stage detectors, such as YOLO [33] and SSD [24], directly predict category confidence and regress bounding box based on predefined anchors. Lin *et al.* [22] proposed focal loss to address class-imbalance, which increases the accuracy of one-stage detector. In this work, we choose Faster R-CNN as baseline detector for its robustness and scalability.

**Unsupervised Domain Adaptation (UDA).** UDA aims to generalize the model learned from labeled source domain to the other unlabeled target domain. In the field of UDA, a group of approaches focus on minimizing a specific domain discrepancy metric, *e.g.*, Maximum Mean Discrepancy (MMD) [11, 44], Weighted MMD [49], Multi-Kernel MMD [25] and Wasserstein Distance [41]. Another research line is based on adversarial training, in which a domain classifier is introduced to facilitate domain-invariance on feature level [7, 43, 26] or pixel level [40, 15, 47]. Recently, several works [46, 54, 30, 3] utilize pseudo labels of samples from target domain to introduce discriminative information during domain alignment. Following the prototype-based approaches [46, 30], we extend the usage of prototype to cross-domain detection tasks.

**Cross-domain Detection.** Beginning with the work of Chen *et al.* [5], the topic of cross-domain detection arouses interests in the community of UDA. In that work, a Domain Adaptive Faster R-CNN model is constructed to reduce domain discrepancy on both image and instance levels. More recently, Saito *et al.* [38] proposed a strong-weak alignment strategy which puts less effort on aligning globally dissimilar images. Cai *et al.* [2] remolded the mean teacher scheme for cross-domain detection. Kim *et al.* [18] used domain diversification to learn feature representations which are invariant among multiple domains. Zhu *et al.* [55] solved the questions of "where to look" and "how to align" via two key components, region mining and region-level alignment. In [17], domain adaptation problem is tackled from the perspective of robust learning.

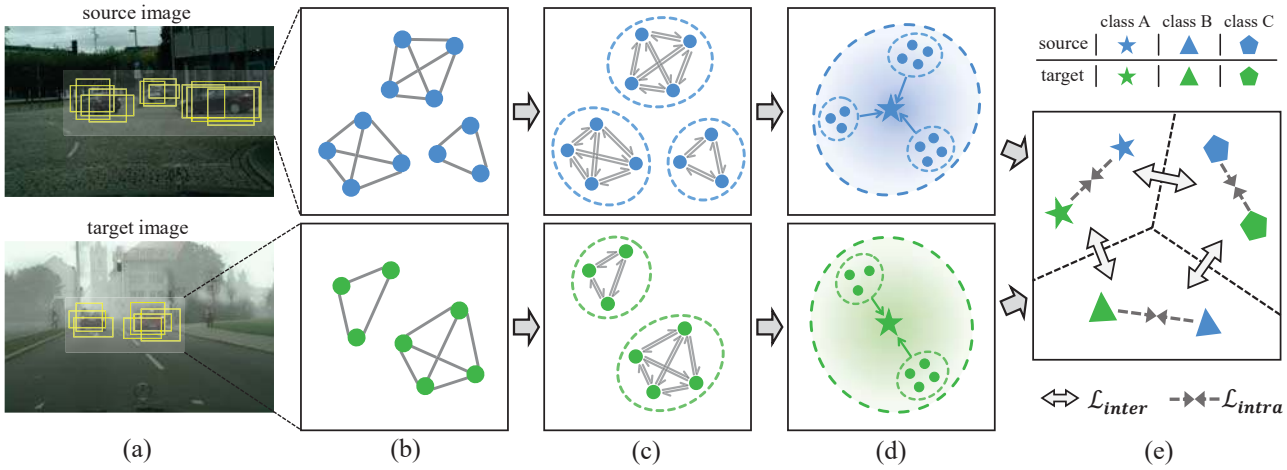



Figure 2. **Framework overview.** (a) Region proposals are generated. (b) Constructing the relation graph on produced region proposals. (c) More accurate instance-level feature representations are obtained through information propagation among proposals belonging to the same instance. (d) Prototype representation of each class is derived via confidence-guided merging. (e) Performing category-level domain alignment through enhancing intra-class compactness and inter-class separability.

*Improvements over existing methods.* Although former works [5, 2, 55] seek for instance-level domain alignment using region proposals, they fail to derive exact instance-level representations and ignore the multi-modal information of various instances. In this work, we utilize relation graph to obtain more precise instance-level feature representations, and per-category prototypes are derived to integrate different instances' multi-modal information.

**Graph Convolutional Network (GCN).** GCN [19] has been explored as a manner to learn graph relations with convolution, which boosts the optimization of graph-based model. Because of the effectiveness and interpretability of GCN, it has been widely applied to various tasks, *e.g.*, action recognition [50], person Re-ID [51], video understanding [45, 52] and point cloud learning [23]. Several recent works [29, 27] utilize graph model to structure multiple domains and categories for classification-based domain adaptation. For cross-domain detection, we employ graph structure to model the relation among region proposals.

## 3. Method

In Unsupervised Domain Adaptation (UDA), source domain $\mathcal{S} = \{(x_i^{\mathcal{S}}, y_i^{\mathcal{S}})\}_{i=1}^{N_{\mathcal{S}}}$ is characterized by $N_{\mathcal{S}}$ i.i.d. labeled samples, where $x_i^{\mathcal{S}}$ follows source distribution $\mathbb{P}_{\mathcal{S}}$ and $y_i^{\mathcal{S}}$ denotes its corresponding label. Similarly, target domain $\mathcal{T} = \{x_j^{\mathcal{T}}\}_{j=1}^{N_{\mathcal{T}}}$ is represented by $N_{\mathcal{T}}$ i.i.d. unlabeled samples, where $x_j^{\mathcal{T}}$ follows target distribution $\mathbb{P}_{\mathcal{T}}$.

### 3.1. Motivation and Overview

In contrast to domain adaptation in classification, its application in object detection is more sophisticated. In specific, since supervisory signal is lacked on target domain, foreground instances are normally represented by a bunch of inaccurate region proposals. In addition, different instances in various scenes commonly reflect diverse modal information, which makes it harder to align source and target domain on local instance level. Another problem impairing model's performance on cross-domain detection tasks is class-imbalance. Concretely, those categories with abundant samples are trained more sufficiently, thus better aligned, while the sample-scarce categories can't be readily aligned for the lack of adaptation training.

To address above issues, we propose the *Graph-induced Prototype Alignment* (GPA) framework. In specific, domain adaptation is realized via aligning two domains' prototypes, in which the critical information of each instance is aggregated via graph-based message propagation, and the multi-modal information reflected by different instances is integrated into per-category prototypes. On the basis of this framework, *Class-imbalance-aware Adaptation Training* is proposed to harmonize the domain adaptation process among different classes through assigning higher weights to the sample-scarce categories.

### 3.2. Graph-induced Prototype Alignment

In the proposed framework, five steps are performed to align source and target domain with category-level prototype representations, just as shown in Figure 2.

**Region proposal generation.** In Faster R-CNN [35], region proposals are generated by Region Proposal Network (RPN) to characterize foreground and background. These proposals provide abundant information of various instance patterns and scene styles, while they usually contain incomplete information of instances because of the deviation of bounding boxes, especially on target domain. Subsequent operations aim to extract the exact information of each instance from region proposals.

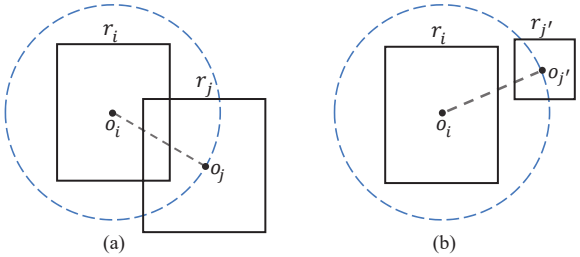



Figure 3. Region proposal $r_i$ interacts with another two region proposals, $r_j$ and $r_{j'}$, with different sizes.

**Constructing relation graph.** We structure the proposals generated by RPN as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ represents the set of vertices corresponding to $N_p$ proposals, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denotes the set of edges, *i.e.* the relations between proposals. Adjacency matrix $\mathbf{A} \in \mathbb{R}^{N_p \times N_p}$ is used to model such relationship. Intuitively, two spatially closer proposals more likely depict the same object and should be assigned with higher connection weight. Following this intuition, a manner to obtain adjacency matrix is to apply a Gaussian kernel over the Euclidean distance between the centers of two proposals:

$$\mathbf{A}_{i,j} = \exp\Big( - \frac{||o_i - o_j||_2^2}{2\sigma^2} \Big), \tag{1}$$

where $o_i$ and $o_j$ denote the centers of the $i$-th and $j$-th proposal ($1 \leqslant i, j \leqslant N_p$), and $\sigma$ is the standard deviation parameter which controls the sparsity of $\mathbf{A}$.

However, when calculating the adjacency matrix, it is unreasonable to treat proposals with various spatial sizes equally. Just as shown in Figure 3, though region proposal pairs $(r_i, r_j)$ and $(r_i, r_{j'})$ have the equal center distance, their strength of relevance is obviously distinct, and $(r_i, r_j)$ should possess higher connection weight in $\mathbf{A}$ for the larger overlap between $r_i$ and $r_j$. Intersection over Union (IoU) is a broadly used metric which takes both the location and size of proposals into consideration, and the derivation of adjacency matrix with IoU is as follows:

$$\mathbf{A}_{i,j} = \mathrm{IoU}(r_i, r_j) = \frac{r_i \bigcap r_j}{r_i \bigcup r_j}, \tag{2}$$

where $r_i$ and $r_j$ denote the $i$-th and $j$-th region proposal respectively ($1 \leqslant i, j \leqslant N_p$). The setup of relation graph lays the foundation for information propagation among region proposals. The comparison between above two methods of constructing adjacency matrix is presented in Sec. 5.1.

**Graph-based region aggregation.** Because of the deviation of bounding boxes, region proposals often distribute around the ground truth objects, which leads to the inaccuracy of representing an object with single proposal. In fact, primal region proposals express incomplete information of instances. In order to achieve exact instance-level feature representations, the embeddings of proposals belonging to a certain instance should be aggregated. By utilizing the spatial relevance provided by adjacency matrix $\mathbf{A}$, proposals' feature embeddings $\mathbf{F} \in \mathbb{R}^{N_p \times d}$ ($d$ is the dimension of embedding) and classification confidence $\mathbf{P} \in \mathbb{R}^{N_p \times N_c}$ ($N_c$ is the number of classes) are aggregated as follows:

$$\widetilde{\mathbf{F}} = \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \mathbf{F}, \tag{3}$$

$$\widetilde{\mathbf{P}} = \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}} \mathbf{P}, \tag{4}$$

where $\mathbf{D} \in \mathbb{R}^{N_p \times N_p}$ denotes the diagonal degree matrix with entries $\mathbf{D}_{ii} = \sum_j \mathbf{A}_{ij}$. In Eqs. 3, 4, after region aggregation, $\widetilde{\mathbf{F}} \in \mathbb{R}^{N_p \times d}$ and $\widetilde{\mathbf{P}} \in \mathbb{R}^{N_p \times N_c}$ express more precise instance-level information through information propagation among adjacent proposals. Compared with the conventional graph convolution, we leave the learnable parameter matrix out, considering that explicit supervisory signal is lacked on the branch of domain adaptation learning. We illustrate the benefit of such operation in Sec. 5.1.

**Confidence-guided merging.** Now that the feature representations are aggregated on instance level, we would like to integrate the multi-modal information reflected by different instances into prototype representations. In order to highlight the modal information which is critical to a specific class, we employ proposals' confidence to each class as the weight during merging, and prototypes are derived as the weighted mean embedding of region proposals:

$$c_k = \frac{\sum_{i=1}^{N_p} \widetilde{\mathbf{P}}_{ik} \cdot \widetilde{\mathbf{F}}_i^T}{\sum_{i=1}^{N_p} \widetilde{\mathbf{P}}_{ik}}, \tag{5}$$

where $c_k \in \mathbb{R}^d$ denotes the prototype of class $k$. The derived prototypes serve as the proxy of each class during subsequent domain alignment.

**Category-level domain alignment.** Prototype-based domain alignment is comprehensively studied in recent literatures [46, 30, 37]. The core idea of these methods is to narrow the distance between same categories' prototypes of two domains, which is achieved through minimizing an intra-class loss, noted as $\mathcal{L}_{intra}$. Furthermore, we propose that the distance between different classes' prototypes should also be constrained with another inter-class loss, noted as $\mathcal{L}_{inter}$. In addition, considering the existence of class-imbalance, the influence of different classes needs to be adjusted. The detailed training scheme is presented in the next section.

### 3.3. Class-imbalance-aware Adaptation Training

In object detection scenario, the class-imbalance problem commonly exists, which means the number of samples belonging to different classes varies greatly. Former work [22] deems that such problem can overwhelm training and degrade detector's performance. In cross-domain detection tasks, class-imbalance can lead to another trouble:

the domain adaptation process among different classes is highly unbalanced. In particular, the feature distributions of sample-scarce categories can't be readily aligned. Inspired by Focal Loss [22] which puts more weights on hard-to-classify examples, we would like to assign higher weights to the sample-scarce categories during the training process of domain adaptation.

Considering that the categories with abundant samples are trained more sufficiently and better aligned, especially in the early training phase, they should possess higher confidence compared with sample-scarce categories. Based on this fact, we select a specific class's highest confidence in a set of proposals, and such confidence value is employed to calculate the weight of this class:

$$p_k = \max_{1 \leqslant i \leqslant N_p} \{\widetilde{\mathbf{P}}_{ik}\}, \tag{6}$$

$$\alpha_k = \begin{cases} (1-p_k)^\gamma & \text{if } p_k > \frac{1}{N_c} \\ 0 & \text{otherwise} \end{cases}, \tag{7}$$

where $p_k$ is the maximum confidence of class $k$ within $N_p$ proposals, and $\gamma$ is the parameter controlling the weights among different classes. Also, we apply a hard threshold, $1/N_c$, to filter out those classes whose samples are not included in the proposal set.

Contrastive loss [12] is commonly used in siamese network architecture to enhance the intra-class compactness and inter-class separability. Utilizing such property, we propose a *Class-reweighted Contrastive Loss* to conduct domain alignment on category level, in which class weights $\{\alpha_i^{\mathcal{S}}\}_{i=0}^{N_c}$ and $\{\alpha_i^{\mathcal{T}}\}_{i=0}^{N_c}$ reweight each term in the loss ("$i = 0$" denotes background). Concretely, in this loss function, the intra-class part requires identical classes' prototypes to be as close as possible, and the inter-class part constrains the distance between different classes' prototypes to be larger than a margin:

$$\mathcal{L}_{intra}(\mathcal{S}, \mathcal{T}) = \frac{\sum_{i=0}^{N_c} \alpha_i^{\mathcal{S}} \alpha_i^{\mathcal{T}} \Phi(c_i^{\mathcal{S}}, c_i^{\mathcal{T}})}{\sum_{i=0}^{N_c} \alpha_i^{\mathcal{S}} \alpha_i^{\mathcal{T}}}, \tag{8}$$

$$\mathcal{L}_{inter}(\mathcal{D}, \mathcal{D}') = \frac{\sum\limits_{0 \leqslant i \neq j \leqslant N_c} \alpha_i^{\mathcal{D}} \alpha_j^{\mathcal{D}'} max(0, m - \Phi(c_i^{\mathcal{D}}, c_j^{\mathcal{D}'}))}{\sum\limits_{0 \leqslant i \neq j \leqslant N_c} \alpha_i^{\mathcal{D}} \alpha_j^{\mathcal{D}'}}, \tag{9}$$

$$\begin{aligned} \mathcal{L}_{da} = \ & \mathcal{L}_{intra}(\mathcal{S}, \mathcal{T}) + \frac{1}{3}\big(\mathcal{L}_{inter}(\mathcal{S}, \mathcal{S}) \\ & + \mathcal{L}_{inter}(\mathcal{S}, \mathcal{T}) + \mathcal{L}_{inter}(\mathcal{T}, \mathcal{T})\big), \end{aligned} \tag{10}$$

where $\Phi(x, x') = ||x - x'||_2$ calculates the Euclidean distance between two prototypes, and $\{c_i^{\mathcal{S}}\}_{i=0}^{N_c}$, $\{c_i^{\mathcal{T}}\}_{i=0}^{N_c}$ denote the prototypes of source and target domain. $\mathcal{D}$ and $\mathcal{D}'$ represent two domains from which pairs of prototypes belonging to different categories are taken. $m$ is the margin term which is fixed as 1.0 in all experiments. In the total domain adaptation loss $\mathcal{L}_{da}$, all pairwise relations between two domains' prototypes are considered.

### 3.4. Two-stage Domain Alignment

Faster R-CNN [35] is a two-stage object detector made up of Region Proposal Network (RPN) and Region-based CNN (R-CNN). First, based on the feature map produced by bottom convolutional layers, RPN generates class-agnostic region proposals. After that, R-CNN predicts fine-grained category labels from feature vectors obtained via ROI pooling. Each stage defines a classification and a localization error, and the total detection loss is defined as follows:

$$\mathcal{L}_{det} = \mathcal{L}_{cls}^{RPN} + \mathcal{L}_{loc}^{RPN} + \mathcal{L}_{cls}^{RCNN} + \mathcal{L}_{loc}^{RCNN}. \tag{11}$$

Based on the two-stage structure of Faster R-CNN, we also conduct domain alignment in a two-stage manner. In the first stage, using the region proposals and corresponding class-agnostic confidence produced by RPN, foreground and background features are separated on latent space, and the foreground feature distributions are aligned as a whole. In the second stage, by utilizing the more accurate bounding boxes and per-category confidence, the feature distribution of each category is respectively aligned. Applying the proposed Class-reweighted Contrastive Loss to both RPN and RCNN, the overall objective is:

$$\min_{F_\theta} \ \mathcal{L}_{det} + \lambda_1 \mathcal{L}_{da}^{RPN} + \lambda_2 \mathcal{L}_{da}^{RCNN}, \tag{12}$$

where $F_\theta$ represents the whole parameterized model, and $\lambda_1$ and $\lambda_2$ are the trade-off parameters between detection and domain adaptation loss.

**Implementation details.** On the basis of ResNet-50 [14] architecture, we implement two domain adaptation losses, $\mathcal{L}_{da}^{RPN}$ and $\mathcal{L}_{da}^{RCNN}$, through adding two domain adaptation learning branches to the $7 \times 7 \times 1024$ feature map after ROI pooling and the 2048-dimensional vector after average pooling, respectively.

## 4. Experiments

In this section, we provide comprehensive experimental results on three cross-domain detection tasks with distinct domain shift, including *Normal to Foggy*, *Synthetic to Real* and *Cross Camera Adaptation*.

### 4.1. Experimental Setup

**Training details.** In all experiments, unless otherwise specified, all of the training and test images are resized such that their shorter side has 600 pixels. During training, for each image, 128 anchors are sampled with a positive to negative ratio of $1:3$. ResNet-50 [14] pre-trained on ImageNet [36] serves as the base architecture. We adopt the SGD optimizer (initial learning rate: 0.001, momentum: 0.9, weight decay: $5 \times 10^{-4}$) to train our model. The number of total training epoch is set as 20, and the learning rate warm-

Table 1. Experimental results (%) of *Normal to Foggy* cross-domain detection task, Cityscapes → Foggy Cityscapes.

| Methods | person | rider | car | truck | bus | train | motorcycle | bicycle | mAP |
|---|---|---|---|---|---|---|---|---|---|
| Source-only | 26.9 | 38.2 | 35.6 | 18.3 | 32.4 | 9.6 | 25.8 | 28.6 | 26.9 |
| DA [5] | 29.2 | 40.4 | 43.4 | 19.7 | 38.3 | 28.5 | 23.7 | 32.7 | 32.0 |
| DivMatch [18] | 31.8 | 40.5 | 51.0 | 20.9 | 41.8 | 34.3 | 26.6 | 32.4 | 34.9 |
| SW-DA [38] | 31.8 | 44.3 | 48.9 | 21.0 | 43.8 | 28.0 | 28.9 | 35.8 | 35.3 |
| SC-DA [55] | **33.8** | 42.1 | 52.1 | **26.8** | 42.5 | 26.5 | 29.2 | 34.5 | 35.9 |
| MTOR [2] | 30.6 | 41.4 | 44.0 | 21.9 | 38.6 | 40.6 | 28.3 | 35.6 | 35.1 |
| GPA (RPN Alignment) | 32.5 | 43.1 | 53.3 | 22.7 | 41.4 | 40.8 | 29.4 | 36.4 | 37.4 |
| GPA (RCNN Alignment) | 33.5 | 44.8 | 52.6 | 26.0 | 41.2 | 37.6 | 29.8 | 35.2 | 37.6 |
| GPA (Two-stage Alignment) | 32.9 | **46.7** | **54.1** | 24.7 | **45.7** | **41.1** | **32.4** | **38.7** | **39.5** |

up strategy [14] is used in the first 200 iterations of training. Without specific notation, the class-balancing hyper-parameter $\gamma$ is set as 2.0, and the IoU-based adjacency matrix defined in Eq. 2 is adopted. For evaluation, we report mean average precisions (mAP) with a threshold of 0.5.

In our experiments, two NVIDIA GeForce 1080 Ti GPUs are used for training, and we select the batch size of 12 to fit GPU memory, *i.e.* 6 images per GPU, consisting of 3 labeled samples from source domain and 3 unlabeled samples from target domain. Our method is implemented with the PyTorch [31] deep learning framework.

**Performance comparison.** We compare our approach with state-of-the-art methods to verify its effectiveness. Our method is evaluated under three configurations corresponding to RPN Alignment ($\lambda_1 = 1.0$, $\lambda_2 = 0.0$), RCNN Alignment ($\lambda_1 = 0.0$, $\lambda_2 = 1.0$) and Two-stage Alignment ($\lambda_1 = 1.0$, $\lambda_2 = 1.0$). Former works, DA [5], DivMatch [18], SW-DA [38], SC-DA [55] and MTOR [2] are introduced for comparison. For the sake of fair comparison, we employ ResNet-50 as the backbone for all these methods. In specific, we re-evaluate the performance of DA, DivMatch, SW-DA and SC-DA using their source code with default configuration, and the performance of MTOR in original paper is reported for the lack of source code.

## 4.2. Normal to Foggy

**Datasets.** In this experiment, Cityscapes [6] and Foggy Cityscapes [39] dataset serve as source and target domain, respectively. Cityscapes dataset contains 2,975 training images and 500 validation images, and we follow the operation in [5] to get the detection annotations. Foggy Cityscapes dataset simulates fog on real scenes through rendering the images from Cityscapes, and it shares the same annotations with Cityscapes dataset. The results are reported on the validation set of Foggy Cityscapes.

**Results.** In Table 1, the comparisons between our approach and other cross-domain detection methods are presented on eight categories. Source-only denotes the baseline Faster R-CNN trained with only source domain data. From the table, it can be observed that the performance of our approach under three configurations all surpasses existing methods. In particular, an increase of $3.6\%$ on mAP is achieved by Two-stage Alignment. The results showcase that, under the domain shift caused by local fog noise, the proposed *graph-based region aggregation* can effectively alleviate such noise and extract critical instance-level features. Take a closer look at per-category performance, our approach achieves highest AP on most sample-scarce categories, *i.e.* rider, bus, train and motorcycle. This phenomenon illustrates the effectiveness of *Class-imbalance-aware Adaptation Training* on balancing the domain adaptation process among different classes.

## 4.3. Synthetic to Real

**Datasets.** In this experiment, SIM 10k [16] dataset is employed as source domain. SIM 10k dataset is collected from the computer game Grand Theft Auto V (GTA5), and it contains 10,000 images. Cityscapes [6] dataset serves as target domain, and experimental results are reported on its validation split.

**Results.** Table 2 reports the performance of our approach compared with other works on two datasets' common category, car. The Two-stage Alignment configuration of our approach obtains the highest AP ($47.6\%$) over all methods. The domain shift of this task is mainly brought by distinct image styles. In such case, in order to achieve satisfactory performance, it's important to produce discriminative features between foreground and background on target domain. We think that, in our framework, such goal is realized through constraining inter-class separability in the *Class-reweighted Contrastive Loss*.

## 4.4. Cross Camera Adaptation

**Datasets.** In this part, we want to explore the adaptation between real-world datasets under different camera setups. KITTI [8] dataset serves as source domain, and it contains 7,481 training images. Cityscapes [6] dataset is utilized as target domain, and its validation set is used for evaluation.

**Results.** The results of various methods on two datasets' common category, car, are presented in Table 3. In this

Table 2. Experimental results (%) of *Synthetic to Real* cross-domain detection task, SIM 10k → Cityscapes.

| Methods | *car* AP |
|---|---|
| Source-only | 34.6 |
| DA [5] | 41.9 |
| DivMatch [18] | 43.9 |
| SW-DA [38] | 44.6 |
| SC-DA [55] | 45.1 |
| MTOR [2] | 46.6 |
| GPA (RPN Alignment) | 45.1 |
| GPA (RCNN Alignment) | 44.8 |
| GPA (Two-stage Alignment) | **47.6** |

Table 3. Experimental results (%) of *Cross Camera Adaptation* task, KITTI → Cityscapes.

| Methods | *car* AP |
|---|---|
| Source-only | 37.6 |
| DA [5] | 41.8 |
| DivMatch [18] | 42.7 |
| SW-DA [38] | 43.2 |
| SC-DA [55] | 43.6 |
| GPA (RPN Alignment) | 46.9 |
| GPA (RCNN Alignment) | 46.1 |
| GPA (Two-stage Alignment) | **47.9** |

task, all three configurations of our approach exceed existing works with a notable margin, in particular, $4.3\%$ performance gain achieved by Two-stage Alignment. In cross camera adaptation tasks, due to the difference of camera setups, abundant patterns exist in instances. In our method, the multi-modal information reflected by various instances is integrated into prototype representations, such that the diverse patterns within a specific category are considered during domain adaptation, which promises the superior performance of our approach.

## 5. Analysis

In this section, we provide more in-depth analysis of our approach to validate the effectiveness of major components with both quantitative and qualitative results.

### 5.1. Ablation Study

**Effect of relation graph.** In Table 4, we analyze a key component, *i.e.* the relation graph, on the task SIM 10k → Cityscapes. The first row directly uses the original region proposals produced by RPN to compute prototypes, and it serves as the baseline. In the second row, we use an Euclidean distance based relation graph defined in Eq. 1, in which $\sigma$ is set as $15.0$ so as to keep the sparsity of derived relation graph same as the one defined by IoU. Comparing the second and fourth row, it can be observed that the configuration using IoU based relation graph performs better, which illustrates that region proposals' size information is essential for relation graph construction.

In the third and fifth row, we append the learnable parameter matrix to Eqs. 3, 4, which forms the conventional formula of graph convolution. After introducing such learnable parameter matrix, compared with the parameter-free counterparts in the second and fourth row, apparent performance decay occurs. We suppose that such phenomenon can be ascribed to the lack of explicit supervisory signal on the branch of domain adaptation learning, which makes it hard to learn a proper feature transformation.

Table 4. Ablation study on different manners to construct relation graph. ("ED": Euclidean distance, "LP": learnable parameter.)

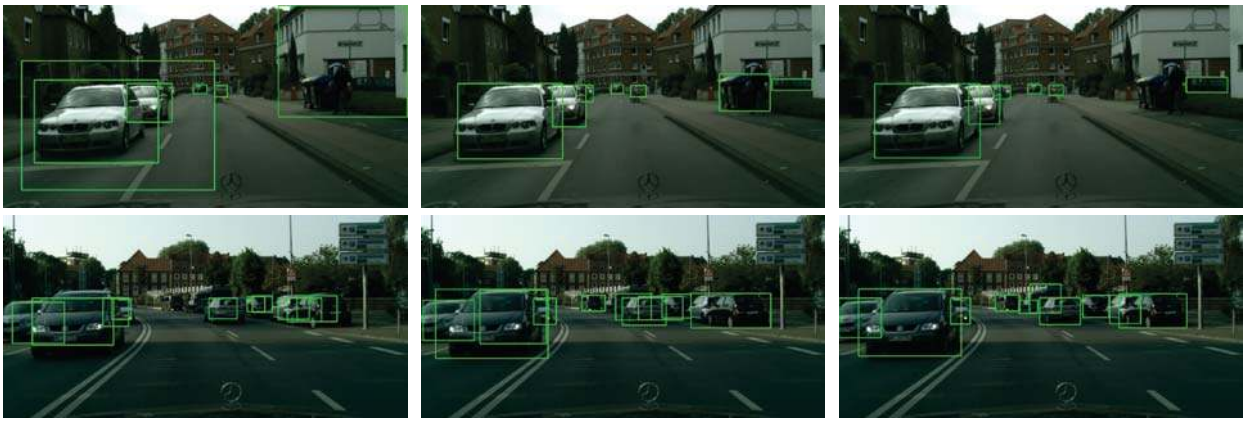
| ED | IoU | LP | *car* AP |
|---|---|---|---|
| | | | 45.0 |
| ✓ | | | 46.1 |
| ✓ | | ✓ | 43.2 |
| | ✓ | | 47.6 |
| | ✓ | ✓ | 43.6 |

**Effect of two-stage alignment.** In this part, we demonstrate the effectiveness of two-stage alignment. In different cross-domain detection tasks, as shown in Table 1, 2 and 3, three configurations of the proposed approach are evaluated. Two single-stage configurations possess similar performance, and two-stage alignment surpasses them with a clear margin. These results illustrate that two-stage alignment boosts domain adaptation via a progressive alignment manner, *i.e.* from coarse-grained foreground alignment to fine-grained per-category alignment.

### 5.2. Sensitivity Analysis

**Sensitivity of trade-off parameters $\lambda_1$, $\lambda_2$.** In this experiment, we validate our approach's sensitivity to $\lambda_1$ and $\lambda_2$ which trade off between detection and domain adaptation loss. Figure 5(a) shows model's performance under different $\lambda_1$ ($\lambda_2$) values when the other parameter $\lambda_2$ ($\lambda_1$) is fixed, and all results are evaluated on the task SIM 10k → Cityscapes. From the line chart, it can be observed that the performance on target domain is not sensitive to both parameters when they vary from $0.25$ to $2.0$, and apparent performance gain is obtained compared with RCNN Alignment ($\lambda_1 = 0$) and RPN Alignment ($\lambda_2 = 0$). This phenomenon illustrates that the two-stage alignment can achieve satisfactory results on a wide range of trade-off parameters.

**Sensitivity of class-balancing parameter $\gamma$.** In this part, we discuss the selection of parameter $\gamma$ which balances the domain adaptation process among different categories. In Figure 5(b), we plot the performance of models trained with different $\gamma$ value on the task Cityscapes →



(a) Source-only (b) DA (c) GPA (Two-stage Alignment)

Figure 4. The detection results on the task SIM 10k → Cityscapes, in which Source-only, DA [5] and our method are evaluated.

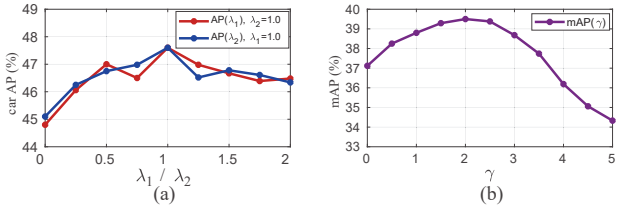



Figure 5. Sensitivity analysis of trade-off parameters $\lambda_1$, $\lambda_2$ (left) and class-balancing parameter $\gamma$ (right).

Foggy Cityscapes. The highest mAP on target domain is achieved when the value of $\gamma$ is around 2.0, which means that, under such condition, the weight assignment among different classes benefits domain adaptation most.

### 5.3. Visualization

**Visualization of two-stage feature.** In Figure 6, we utilize t-SNE [28] to visualize the feature distribution of source and target domain on the task SIM 10k → Cityscapes, in which the feature embeddings of both RPN and RCNN phase are used for visualization. Compared with the Source-only model, after conducting RPN and RCNN alignment, the features of the same category in two domains are better aligned, and different categories' features are separated more clearly. This visually verifies that the proposed method boosts feature alignment on both stages.

**Qualitative detection results.** Figure 4 displays some typical detection results on the task SIM 10k → Cityscapes, in which Source-only, DA [5] and our approach are evaluated. As shown in the figure, the Source-only model can poorly localize objects. DA [5] predicts bounding box more accurately, but it incorrectly classifies the garbage can as a car, and produces some false positives. Our model successfully inhibits false positives, and it is able to localize objects precisely even when severe occlusion occurs.

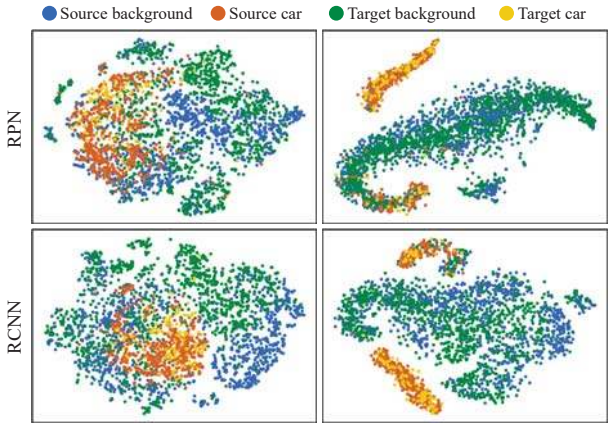



Figure 6. The t-SNE [28] visualization of feature embeddings produced by RPN and RCNN, in which Source-only model and our method are employed for feature extraction.

## 6. Conclusion

In this paper, we propose the Graph-induced Prototype Alignment (GPA) framework for cross-domain detection. In the framework, the critical information of each instance is aggregated through graph-based message propagation, and prototype representations are derived for category-level domain alignment. Furthermore, we harmonize the process of adaptation training through Class-reweighted Contrastive Loss. Extensive experiments and analytical studies demonstrate the prominent performance of our approach.

## 7. Acknowledgement

This work was supported by National Science Foundation of China (61976137, U1611461, U19B2035) and STCSM(18DZ1112300). This work was also supported by National Key Research and Development Program of China (2016YFB1001003). Authors would like to appreciate the Student Innovation Center of SJTU for providing GPUs.

## References


[1] Konstantinos Bousmalis, Nathan Silberman, David Dohan, Dumitru Erhan, and Dilip Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

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

[3] Chaoqi Chen, Weiping Xie, Wenbing Huang, Yu Rong, Xinghao Ding, Yue Huang, Tingyang Xu, and Junzhou Huang. Progressive feature alignment for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[4] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *International Conference on Learning Representations*, 2015.

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

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

[7] Yaroslav Ganin and Victor S. Lempitsky. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning*, 2015.

[8] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[9] Ross B. Girshick. Fast R-CNN. In *IEEE International Conference on Computer Vision*, 2015.

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

[11] Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander J. Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13:723–773, 2012.

[12] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[13] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *IEEE International Conference on Computer Vision*, 2017.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[15] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A. Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International Conference on Machine Learning*, 2018.

[16] Matthew Johnson-Roberson, Charles Barto, Rounak Mehta, Sharath Nittur Sridhar, Karl Rosaen, and Ram Vasudevan. Driving in the matrix: Can virtual worlds replace human-generated annotations for real world tasks? In *IEEE International Conference on Robotics and Automation*, 2017.

[17] Mehran Khodabandeh, Arash Vahdat, Mani Ranjbar, and William G. Macready. A robust learning approach to domain adaptive object detection. In *IEEE International Conference on Computer Vision*, 2019.

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

[19] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2012.

[21] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

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

[23] Jinxian Liu, Bingbing Ni, Caiyuan Li, Jiancheng Yang, and Qi Tian. Dynamic points agglomeration for hierarchical point sets learning. In *IEEE International Conference on Computer Vision*, 2019.

[24] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. In *European Conference on Computer Vision*, 2016.

[25] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I. Jordan. Learning transferable features with deep adaptation networks. In *International Conference on Machine Learning*, 2015.

[26] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I. Jordan. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, 2018.

[27] Xinhong Ma, Tianzhu Zhang, and Changsheng Xu. GCAN: graph convolutional adversarial network for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[28] Laurens Van Der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(2605):2579–2605, 2008.

[29] Massimiliano Mancini, Samuel Rota Bulò, Barbara Caputo, and Elisa Ricci. Adagraph: Unifying predictive and continuous domain adaptation through graphs. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[30] Yingwei Pan, Ting Yao, Yehao Li, Yu Wang, Chong-Wah Ngo, and Tao Mei. Transferrable prototypical networks for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[31] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NeurIPS Workshop*, 2017.

[32] Zhongyi Pei, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. Multi-adversarial domain adaptation. In *AAAI Conference on Artificial Intelligence*, 2018.

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

[34] Joseph Redmon and Ali Farhadi. YOLO9000: better, faster, stronger. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[35] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, 2015.

[36] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[37] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko. Semi-supervised domain adaptation via minimax entropy. In *IEEE International Conference on Computer Vision*, 2019.

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

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

[40] Swami Sankaranarayanan, Yogesh Balaji, Carlos D. Castillo, and Rama Chellappa. Generate to adapt: Aligning domains using generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[41] Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein distance guided representation learning for domain adaptation. In *AAAI Conference on Artificial Intelligence*, 2018.

[42] Baochen Sun and Kate Saenko. Deep CORAL: correlation alignment for deep domain adaptation. In *ECCV Workshop*, 2016.

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

[44] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *CoRR*, abs/1412.3474, 2014.

[45] Xiaolong Wang and Abhinav Gupta. Videos as space-time region graphs. In *European Conference on Computer Vision*, 2018.

[46] Shaoan Xie, Zibin Zheng, Liang Chen, and Chuan Chen. Learning semantic representations for unsupervised domain adaptation. In *International Conference on Machine Learning*, 2018.

[47] Minghao Xu, Jian Zhang, Bingbing Ni, Teng Li, Chengjie Wang, Qi Tian, and Wenjun Zhang. Adversarial domain adaptation with domain mixup. *CoRR*, abs/1912.01805, 2019.

[48] Hongliang Yan, Yukang Ding, Peihua Li, Qilong Wang, Yong Xu, and Wangmeng Zuo. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[49] Hongliang Yan, Yukang Ding, Peihua Li, Qilong Wang, Yong Xu, and Wangmeng Zuo. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[50] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI Conference on Artificial Intelligence*, 2018.

[51] Yichao Yan, Qiang Zhang, Bingbing Ni, Wendong Zhang, Minghao Xu, and Xiaokang Yang. Learning context graph for person search. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[52] Yichao Yan, Ning Zhuang, Bingbing Ni, Jian Zhang, Minghao Xu, Qiang Zhang, Zhang Zheng, Shuo Cheng, Qi Tian, Yi Xu, Xiaokang Yang, and Wenjun Zhang. Fine-grained video captioning via graph-based multi-granularity interaction learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[53] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in Neural Information Processing Systems*, 2014.

[54] Weichen Zhang, Wanli Ouyang, Wen Li, and Dong Xu. Collaborative and adversarial network for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

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