SIGMA: Semantic-complete Graph Matching for Domain Adaptive Object Detection

Wuyang Li Xinyu Liu Yixuan Yuan*
City University of Hong Kong
{wuyangli2, xliu423}-c@my.cityu.edu.hk yxyuan.ee@cityu.edu.hk

Abstract

Domain Adaptive Object Detection (DAOD) leverages a labeled domain to learn an object detector generalizing to a novel domain free of annotations. Recent advances align class-conditional distributions by narrowing down cross-domain prototypes (class centers). Though great success, they ignore the significant within-class variance and the domain-mismatched semantics within the training batch, leading to a sub-optimal adaptation. To overcome these challenges, we propose a novel SemantIc-complete Graph MAtching (SIGMA) framework for DAOD, which completes mismatched semantics and reformulates the adaptation with graph matching. Specifically, we design a Graph-embedded Semantic Completion module (GSC) that completes mismatched semantics through generating hallucination graph nodes in missing categories. Then, we establish cross-image graphs to model class-conditional distributions and learn a graph-guided memory bank for better semantic completion in turn. After representing the source and target data as graphs, we reformulate the adaptation as a graph matching problem, i.e., finding well-matched node pairs across graphs to reduce the domain gap, which is solved with a novel Bipartite Graph Matching adaptor (BGM). In a nutshell, we utilize graph nodes to establish semantic-aware node affinity and leverage graph edges as quadratic constraints in a structure-aware matching loss, achieving fine-grained adaptation with a node-to-node graph matching. Extensive experiments verify that SIGMA outperforms existing works significantly. Our code is available at https://github.com/CityU-AIM-Group/SIGMA.

1. Introduction

Well-trained object detectors [23, 24, 33] have been proven to achieve promising performance with a consistent distribution of training and test data. However, deploying these methods in a novel domain leads to the catastrophic performance degradation due to the domain gap [3], which significantly limits the generalization and transferability of object detectors. Furthermore, this challenge also restricts the application of object detection in real-world scenarios, such as self-driving under distinctive weather conditions and video analysis containing novel scenes.

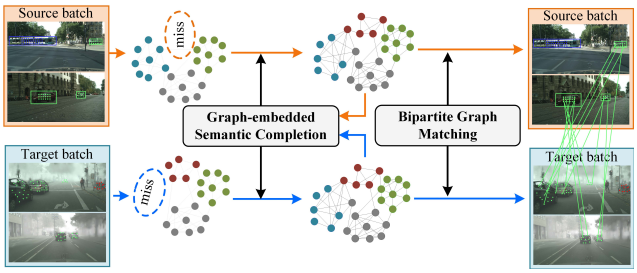

Figure 1. Illustration of the proposed SemantIc-complete Graph MAtching (SIGMA) framework for DAOD.

To overcome this limitation, Unsupervised Domain Adaptation (UDA) methods have been explored to adapt the unlabeled target domain and the annotated source domain, and one of the main streams of UDA works is to align feature distributions between source and target domains. Early works [3, 12, 27] adopt a pixel-to-pixel adaptation in terms of hierarchical features, yielding a global alignment of the whole image with per-pixel adaptation. Some works [3, 37, 41] focus on foreground objects and conduct more precise adaptation on those regions of interest. Recently, some works [32,37,41,42] aim to align cross-domain class-conditional distributions in the implicit feature space and achieve adaptation in a category-to-category manner. These works model category centers with prototypes and minimize the distance of cross-domain prototypes to bridge the domain gap at the category level.

Though satisfactory performance, there are still two challenges in existing category-level adaptation works [32, 37, 41, 42]. Firstly, these works neglect the significant within-class variance and directly align handcraft category

*Yixuan Yuan is the corresponding author.
This work was supported by Hong Kong Research Grants Council (RGC) General Research Fund 11211221 (CityU 9043152).

centers, which inevitably bring about a sub-optimal adaptation. Due to the diverse size and appearance of object instances, the within-class variance covers essential information to represent class-conditional distributions, e.g., the scale and shape, which should also be aligned for domain adaptation. Overlooking the within-class variance could lead to lots of non-adapted object instances and the potential overlapping of different class-conditional distributions with false-positive classification errors. Although some works have introduced explicit variance [2] to relieve the problem of existing center-based measurements, they follow the Gaussian assumption to model feature distributions, which is not optimal in the non-convex deep feature space. These observations motivate us to design a new paradigm to align cross-domain pixel-pairs in the non-euclidean graphical space [39], which models and adapts class-conditional distributions without handcraft center-based alignment.

The second challenge lies in the domain-mismatched semantics within the training batch. Some existing works [32, 37, 42] only perform adaptation on the co-occurred categories in two domains, ignoring mismatched categories appearing in a single domain. Neglecting missing categories leads to a non-effective adaptation due to the loss of semantic knowledge. As shown in Figure 1, the train only appears in the source batch, while these bicycles are available in the target domain, yielding inconsistent semantics across domains. These mismatched semantics bring about the difficulty of explicitly estimating class centers, limiting the adaptation of class-conditional distributions. Furthermore, the missing semantics in the target domain even result in the potential risk towards source-specific direction since the supervised source classification could generate a biased class-conditional distribution [35]. Hence, we are committed to designing a semantic completion strategy through generating novel hallucination samples [40] in the missing categories, which relieves the negative impact of mismatched semantics and achieves more effective adaptation.

To overcome the aforementioned challenges, we propose a SemantIc-complete Graph MAtching (SIGMA) framework for DAOD, which completes domain-mismatched semantics and reformulates the adaptation as a graph matching problem, i.e., finding the suitable matching between graph nodes to bridge the domain gap. As shown in Figure 1, we design a Graph-embedded Semantic Completion module (GSC) to complete the mismatched semantics, which utilizes domain-level statistics to generate hallucination nodes in the missing categories. Then, we establish graphs to model class-conditional distributions for both domains and learn a graph-guided memory bank to improve the capacity of semantic completion in turn. Based on our reformulation of domain adaptation, we propose a Bipartite Graph Matching adaptor (BGM) to solve the graph matching problem between the source and target graph, achieving a fine-grained domain alignment. We utilize graph nodes to learn semantic-aware node affinity and introduce graph edges in a structure-aware matching loss for the Quadratic Assignment Problem (QAP). This graph-matching-based domain alignment enables a fine-grained adaptation with well-matched semantics and relieves the biased and non-effective adaptation in existing prototype-based methods. To be summarized, our contributions are as follows.

- We propose a SemantIc-complete Graph MAtching (SIGMA) framework for DAOD, which aligns the class-conditional distribution with graph matching. To the best of our knowledge, this work represents the first attempt to leverage graph matching theory to bridge the domain gap in the detection community.

- We propose a Graph-embedded Semantic Completion module (GSC) to complete mismatched semantics by generating hallucination nodes and a Bipartite Graph Matching adaptor (BGM) that reformulates DAOD as a graph matching problem to bridge the domain gap.

- Extensive experiments on three benchmarks demonstrate that SIGMA achieves state-of-the-art results and outperforms DAOD counterparts significantly.

2. Related Work

2.1. Domain Adaptive Object Detection

Domain adaptive object detection (DAOD) aims to bridge the domain gap between the training and testing data, which can be categorized into style-transfer [13, 14, 16], self-labeling [14, 21], and domain-alignment [3, 17, 27]. As one of the main streams, domain-alignment approaches adopt adversarial feature alignment and minimize the cross-domain discrepancy to bridge the domain gap. Early works align global features [3, 17, 27] with diverse mechanisms, e.g., spatial attention [17] and strong-weak alignment [27]. Besides, some works tend to align a community of local pixels with essential attributes, e.g., region proposals [16] and object centers [12]. Recently, some works have introduced a more precise adaptation in class-conditional distributions at the category level. GPA [37] utilizes RoI-based graphs to model prototypes and narrows down these cross-domain measurements. PARPN [41] extends the idea of prototype alignment in the RPN stage, and the authors in [42] extend the batch-wise prototypes at the domain level. However, these works ignore the significant within-class variance, leading to a sub-optimal alignment of class-conditional distributions. This work breaks this barrier with graph matching, avoiding the inaccuracy adaptation caused by handcraft prototype design and center-based alignment.

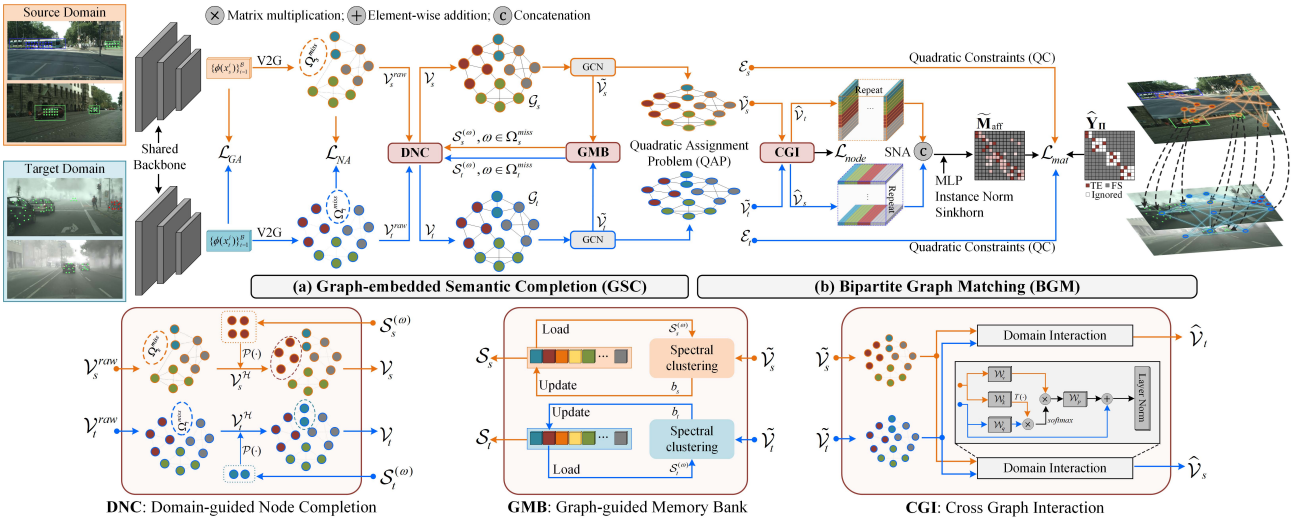

Figure 2. Overview of the proposed SIGMA framework for DAOD. V2G represents vision-to-graph transformation.

2.2. Graph Matching

Graph matching establishes pair-wise node correspondences between two graphs, and gives a one-to-one matching of graph nodes belonging to different graphical entities. As a Quadratic Assignment Problem (QAP) [20] with combinational nature, graph matching solvers [20, 38] optimize a cross-graph permutation matrix to encode matched node pairs, considering both node and structure affinities. Recently, graph matching has been extended to visual correspondence detection [8], multi-object tracking [10], point cloud registration [6] to model pair-wise relationships in the graphical space. Gao, *et al*. [8] model key-point-based graphs on images and establish graph matching between images covering the same objects. Fu *et al*. model graphs on the 3D rigid point cloud and perform graph matching on two homogeneous point sets to achieve robust point cloud registration. The authors in [10] perform graph matching across the tracklet and detection space to achieve high-quality object tracking. Different from aforementioned scenarios with off-the-shelled graph definition and pair-wise labels, we innovatively reformulate DAOD as a graph matching problem, and leverage the QAP solver to bridge the domain gap.

3. Motivation and Preliminaries

We theoretically analyze existing category-level adaptation approaches, and demonstrate our motivation and new solution as follows. Considering the batch-wise source and target observation $\mathcal{S} = \{(x_s^i, y_s^i)\}_{i=1}^{\mathcal{B}}$ and $\mathcal{T} = \{x_t^i\}_{i=1}^{\mathcal{B}}$ drawn from the inconsistent domain distribution $\mathcal{P}_s$ and $\mathcal{P}_t$ ($\mathcal{P}_s \neq \mathcal{P}_t$), existing approaches [32, 37, 41, 42] aim to model and align class-conditional distributions $\mathcal{P}_{X|Y}(\phi(x_{s/t})|y)$, where $\phi(\cdot)$ is the feature extractor. These works first estimate category centers $\mu_{s/t}^y = \mathbb{E}_{X|Y}[\phi(x)|y]$ with handcraft priors, e.g., mean-values of object features $\mu_{s/t}^y = \frac{1}{N_{s/t}} \sum_i^{N_{s/t}} RoI_i^y$, and then minimize the domain-discrepancy between μ_s^y and μ_t^y. However, these methods potentially achieve a biased adaptation depending only on center-based knowledge, and fail to adapt mismatched categories $\Omega_{s/t}^{miss}$ appearing in a single domain due to the intractable $\mu_{s/t}^{y=\Omega_{s/t}^{miss}}$.

To overcome these issues, we generate novel samples in the missing categories $\Omega_{s/t}^{miss}$ to complete the mismatched semantic, and establish a cross-image graph $\mathcal{G}_{s/t}$ to model the class-conditional distribution $\mathcal{P}_{X|Y}(\phi(x_{s/t})|y)$ for each domain. Then, we reformulate domain adaptation as a graph matching problem between $\mathcal{G}_s$ and $\mathcal{G}_t$, which can be solved with a differential QAP [6, 8, 10] as follows,

$$\begin{aligned} &\min_{\mathbf{\Pi}} \mathcal{F}(\mathbf{\Pi}) = ||\mathcal{A}_s - \mathbf{\Pi}\mathcal{A}_t\mathbf{\Pi}^T||_F^2 - tr(\mathbf{X}_u^T\mathbf{\Pi}), \\ &\mathbf{\Pi} \in [0,1]^{\mathcal{N}_s \times \mathcal{N}_t}, \mathbf{\Pi}\mathbf{1}_{\mathcal{N}_s} \leq \mathbf{1}_{\mathcal{N}_t}, \mathbf{\Pi}^T\mathbf{1}_{\mathcal{N}_t} \leq \mathbf{1}_{\mathcal{N}_s}, \end{aligned} \tag{1}$$

where $\mathcal{A}_s \in \mathbb{R}^{\mathcal{N}_s \times \mathcal{N}_s}$ and $\mathcal{A}_t \in \mathbb{R}^{\mathcal{N}_t \times \mathcal{N}_t}$ represent the adjacent matrix encoding structure information of the graph $\mathcal{G}_s$ and $\mathcal{G}_t$ respectively, $\mathcal{N}_{s/t}$ is the number of graph nodes, $||\cdot||_F$ is the Frobenius norm, $\mathbf{X}_u \in \mathbb{R}^{\mathcal{N}_t \times \mathcal{N}_s}$ is the unary affinity matrix and generally specified as the node affinity $\mathbf{M}_{\text{aff}}$ [8], and $\mathbf{\Pi}$ is the relaxed permutation matrix encoding node-to-node assignment [1] and $\mathbf{\Pi}_{i,j} = 1$ indicates that the node $v_s^i \in \mathcal{G}_s$ is matched with the node $v_t^j \in \mathcal{G}_t$.

Different from existing works [37, 41, 42] overlooking mismatched categories, we complete missing semantics and effectively align the distribution for each appeared category. Besides, our method achieves a fine-grained adaptation guided by graph matching, breaking the barrier of existing center-based methods adopting sub-optimal alignment.

[1] We follow [8] to relax the one-hot permutation matrix with continuous values to satisfy the differential requirement of neural network training.

4. Proposed Method

The overall workflow the proposed SIGMA framework is shown in Figure 2. Given batch-wise annotated source images $\{(x_s^i, y_s^i)\}_{i=1}^{\mathcal{B}}$ and unlabeled target images $\{x_t^i\}_{i=1}^{\mathcal{B}}$, we use a shared feature extractor ϕ to extract image-level features $\{\phi(x_{s/t}^i)\}_{i=1}^{\mathcal{B}}$, which are sent to Graph-embedded Semantic Completion module (GSC) (Figure 2(a)). In the GSC module, we first transform visual features to the graphical space (V2G) and perform domain-guided node completion (DNC) to complete mismatched semantics, obtaining semantic-complete node sets $\mathcal{V}_{s/t}$. Then, we establish cross-image graphs $\mathcal{G}_{s/t}$ to model the class-conditional distribution with enhanced nodes $\tilde{\mathcal{V}}_{s/t}$, which also serves to learn a graph-guided memory bank (GMB) to improve the semantic completion in turn. Afterwards, the well-modeled graphs $\mathcal{G}_{s/t}$ are sent to the Bipartite Graph Matching adaptor (BGM) (Figure 2(b)). We use graph nodes $\tilde{\mathcal{V}}_{s/t}$ for cross graph interaction (CGI) and learn a semantic-aware node affinity (SNA) matrix $\tilde{\mathbf{M}}_{\text{aff}}$. Besides, we leverage graph edges $\mathcal{E}_{s/t}$ to serve as quadratic constraints (QC) to optimize the graph matching permutation, achieving fine-grained adaptation with well-aligned node pairs.

4.1. Graph-embedded Semantic Completion

Given batch-wise annotated source images $\{(x_s^i, y_s^i)\}_{i=1}^{\mathcal{B}}$ and unlabeled target images $\{x_t^i\}_{i=1}^{\mathcal{B}}$ with C categories, we first adopt the domain-shared backbone ϕ to extract visual features $\{\phi(x_{s/t}^i)\}_{i=1}^{\mathcal{B}}$, $\phi(x_{s/t}^i) \in \mathbb{R}^{D\times W\times H}$. For the source features, we perform spatial-uniformed sampling to collect the pixels inside ground-truth boxes as class-aware foreground nodes and a ratio $\frac{1}{C+1}$ of pixels outside foreground boxes as background samples. For the target domain, we forward-propagate target features in classification head to obtain pseudo score maps $\mathcal{M}_t \in \mathbb{R}^{C\times W\times H}$ as the surrogate sampling principle. Then we sample the pixels satisfying $\max_C(\mathcal{M}_t^i) > \tau_{fg}$ as class-aware foreground nodes and a ratio $\frac{1}{C+1}$ of low-score pixels ($\max_C(\mathcal{M}_t^i) < \tau_{bg}$) as background samples[2]. After sampling fine-grained visual features, we perform a non-linear projection to obtain the raw node embedding $\mathcal{V}_{s/t}^{raw} = \{v_{s/t}^i\}_{i=1}^{\mathcal{N}_{s/t}}$, achieving the transformation from the visual space to the graphical space.

Domain-guided Node Completion. The object categories $\Omega_{s/t}^{\mathcal{B}} \in \{0, 1, ..., C\}$ within a training batch are always mismatched between the source and target domain, limiting the adaptation of class-conditional distributions. Hence, we propose a semantic completion strategy to generate hallucination nodes in missing categories $\Omega_s^{miss} = \{\omega|\omega \in \Omega_t^{\mathcal{B}}, \omega \notin \Omega_s^{\mathcal{B}}\}$, $\Omega_t^{miss} = \{\omega|\omega \in \Omega_s^{\mathcal{B}}, \omega \notin \Omega_t^{\mathcal{B}}\}$, obtaining semantic-complete nodes $\mathcal{V}_{s/t}$. To generate additional nodes containing non-existing semantics, we define a graph-guided memory bank $\mathcal{S}_{s/t} \in \mathbb{R}^{C\times D}$ to save the category-specific knowledge of inner-domain semantics, and we will explain the learning strategy of this memory bank in the next section. Considering the source and target domains share a similar category space [3], we fully utilize the semantic cues from the counterpart domain to guide the node generation, which provide a joint measurement of the class-conditional distribution within the batch. Specifically for the completion of the source-missing category $\omega \in \Omega_s^{miss}$, we calculate the standard variance of target nodes $\{v_t^{(\omega)}\}$ in class ω to obtain a variant vector $\sigma_t^{(\omega)} \in \mathbb{R}^D$, which approximates the scale of the distribution for the missing category ω. Then, we load the corresponding memory seed $\mathcal{S}_s^{(\omega)}$ from the memory bank to serve as the category-specific expectation $\mu_s^{(\omega)}$. After that, we perform Gaussian sampling and adopt a linear projection $\mathcal{P}(\cdot)$ to obtain hallucination nodes $\mathcal{V}_s^{\mathcal{H}} = \{v_s^h | v_s^h = \mathcal{P}(x_s^h), x_s^h \sim N(\mu_s^{(\omega)}, \sigma_t^{(\omega)})\}$ belonging to the mismatched categories. The same completion is also conducted in the target domain to obtain the nodes $\mathcal{V}_t^{\mathcal{H}}$ in the target-missing categories Ω_t^{miss}. Instead of aligning these statistic-based estimations directly [37, 41, 42], we fully utilize domain knowledge to generate novel and unbiased samples, avoiding the biased and sub-optimal alignment. Finally, both existing nodes and hallucination ones constitute the semantic-complete node set $\mathcal{V}_{s/t}$ for the followed graph modelling.

Graph-guided Memory Bank. Since the nodes $\mathcal{V}_{s/t}$ derive from different images within a batch, we establish a cross-image graph to model the class-conditional distribution with long-distance semantic dependency, and propose a memory bank to preserve graph-based knowledge, which helps the DNC to generate better hallucination nodes in turn. Specifically, we first introduce edge connections $\mathcal{E}_{s/t}$ between nodes $\mathcal{V}_{s/t}$ and set up a cross-image graph $\mathcal{G}_{s/t} = \{\mathcal{V}_{s/t}, \mathcal{E}_{s/t}\}$ in each domain. For the graph edge, we utilize edge drop [26] to avoid the potential relationship bias caused by the abundant visual correspondence: $\mathcal{A}_{s/t} = Edgedrop\{softmax[\mathcal{V}_{s/t}\mathcal{W}_e(\mathcal{V}_{s/t}\mathcal{W}_e)^T]\}$, where $\mathcal{A}_{s/t}$ is the adjacent matrix encoding structure information, and $\mathcal{W}_e$ is a learnable linear projection. Then, we perform single-layer graph convolution with the graph-based message propagation among nodes to aggregate cross-image semantic knowledge, yielding the enhanced node representation: $\tilde{v}_{s/t}^i = \text{LN}(\sum_{v_{s/t}^j \in \mathcal{NR}^i}^{|\mathcal{NR}^i|} \mathcal{A}_{s/t}^{i,j} v_{s/t}^j \mathcal{W}_{gcn} + v_{s/t}^i)$, where $\mathcal{NR}^i$ represents the neighbour nodes of $v_{s/t}^i$, $\mathcal{W}_{gcn}$ is the learnbale parameter, and LN is the layer normalization [1].

To provide representative and robust dependency for the hallucination node generation, we introduce a memory bank to save class-specific graph embedding and design a cluster-based update strategy for the memory bank learning. Specifically, we randomly initialize a memory bank

[2] τ_{fg} is empirically set 0.5 to satisfy the active condition of the non-linear *sigmoid* function and τ_{bg} is set 0.05 following the commonly used score-threshold setting in existing object detectors [19, 23, 24, 33].

$\mathcal{S}_{s/t} \in \mathbb{R}^{C\times D}$ at the beginning of the training and gradually update memory seeds with appeared graph nodes. For each appeared category ω within a training batch, we collect graph nodes $\{\tilde{v}^{(\omega)}_{s/t}\}, \tilde{v}^{(\omega)}_{s/t} \in \mathbb{R}^D$ in class ω and load the corresponding memory seed $\mathcal{S}^{(\omega)}_{s/t} \in \mathbb{R}^D$ from the memory bank $\mathcal{S}_{s/t}$. Then, we get both the memory seed and graph nodes together $\{\mathcal{S}^{(\omega)}_{s/t}, \tilde{v}^{(\omega)}_{s/t}\}$ and conduct spectral clustering [31] in the graphical space to generate two clusters, i.e., a seed-included cluster $\pi^{seed}_{s/t} = \{\mathcal{S}^{(\omega)}_{s/t}, \tilde{v}^{(\omega)}_{s/t}\}$ and an "else" cluster $\pi^{else} = \{\tilde{v}^{(\omega)}_{s/t}\}$. Since the domain-level knowledge, referred to as the memory seed, provides a more robust and precise estimation compared with the batch-wise measurement, we only utilize the nodes in $\pi^{seed}_{s/t}$ to update the memory bank, which relieves the impact of noisy nodes appeared in the early training stage:

$$\mathcal{S}^{(\omega)}_{s/t} \leftarrow sim(b_{s/t}, \mathcal{S}^{(\omega)}_{s/t})\mathcal{S}^{(\omega)}_{s/t} + [1 - sim(b_{s/t}, \mathcal{S}^{(\omega)}_{s/t})]b_{s/t}, \tag{2}$$

where $sim(b_{s/t}, \mathcal{S}^{(\omega)}_{s/t}) = \frac{b_{s/t}\cdot\mathcal{S}^{(\omega)}_{s/t}}{\|b_{s/t}\|_2 \cdot \|\mathcal{S}^{(\omega)}_{s/t}\|_2}$ indicates the adaptive momentum for better gradient-free learning [34, 42], and $b_{s/t} = \frac{1}{|\pi^{seed}_{s/t}|-1}\sum_{\tilde{v}^{(\omega)}_{s/t}\in\pi^{seed}_{s/t}} \tilde{v}^{(\omega)}_{s/t}$. We only utilize existing graph nodes to update memory seeds, and remove those hallucination ones to avoid the potential negative impact of handcraft Gaussian priors for the model learning.

4.2. Bipartite Graph Matching

Given the graph $\mathcal{G}_{s/t}$, we reformulate the cross-domain alignment as a graph matching problem, i.e., solving the QAP between $\mathcal{G}_s$ and $\mathcal{G}_t$. Specifically, we use graph nodes $\tilde{\mathcal{V}}_{s/t}$ to establish cross-graph interaction and learn a node affinity $\tilde{\mathbf{M}}_{\text{aff}}$. Besides, we introduce graph edges $\mathcal{E}_{s/t}$ to bridge the domain gap with a structure-aware matching loss.

Cross Graph Interaction. Since graph matching is a collaborative optimization problem between two graphical entities, the message propagation across graphs is essential for the optimal solution in graph-based affinity learning. Hence, we introduce the knowledge exchange between $\mathcal{G}_s$ and $\mathcal{G}_t$ to establish the cross-domain semantic interaction:

$$\begin{aligned}\hat{\mathcal{V}}_s &= \text{LN}\{softmax[(\tilde{\mathcal{V}}_s\mathcal{W}_q)(\tilde{\mathcal{V}}_t\mathcal{W}_k)^T](\tilde{\mathcal{V}}_t\mathcal{W}_v)\mathcal{W}_p + \tilde{\mathcal{V}}_s\},\\ \hat{\mathcal{V}}_t &= \text{LN}\{softmax[(\tilde{\mathcal{V}}_t\mathcal{W}_q)(\tilde{\mathcal{V}}_s\mathcal{W}_k)^T](\tilde{\mathcal{V}}_s\mathcal{W}_v)\mathcal{W}_p + \tilde{\mathcal{V}}_t\},\end{aligned} \tag{3}$$

where $\hat{\mathcal{V}}_{s/t} = \{\hat{v}^i_{s/t}\}^{\mathcal{N}_{s/t}}_{i=1}$ is the graph node set with cross-domain perception, LN is the layer normalization [1], and $\mathcal{W}_{(\cdot)}$ are learnable parameters. To enhance the graphical semantics, we introduce an auxiliary node classification task by adopting a classifier f_{cls} with the Cross Entropy loss:

$$\mathcal{L}_{node} = -\sum_{i=1}^{\mathcal{N}_s+\mathcal{N}_t} y_i log\{softmax[f_{cls}(\hat{v}^i_{s/t})]\}, \tag{4}$$

where y_i represents the ground-truth label for source nodes and the pseudo label (obtained from score maps $\mathcal{M}_t$) for target nodes. Dense relationships can be established among nodes belonging to different domains, serving the sparse and fine-grained adaptation with interactive semantic cues.

Semantic-aware Node Affinity. Given the graph nodes $\hat{\mathcal{V}}_{s/t}$ with cross-domain perception, we further learn an affinity matrix to model the node correspondence between $\mathcal{G}_s$ and $\mathcal{G}_t$. Different from existing graph matching approaches [6, 8, 10] utilizing local visual representations, we leverage the category-level semantic with inherent relationships to learn a semantic-aware affinity matrix. Specifically, we define the entry of the node affinity matrix as follows: $\mathbf{M}^{i,j}_{\text{aff}} = f_{mlp}\{f_p(\hat{v}^i_s) \copyright f_p(\hat{v}^j_t)\}$, $\mathbf{M}_{\text{aff}} \in \mathbb{R}^{\mathcal{N}_s\times\mathcal{N}_t}$, where $\copyright$ is the concatenation operation, f_p indicates a linear projection, and f_{mlp} is a multi-layer perceptron layer (MLP) with a single output channel. This MLP layer learns inherent semantic relationships between two graph nodes and encodes them into affinity representations. $\mathbf{M}_{\text{aff}}$ is then sent to the Instance Normalization layer as [6] and the differential Sinkhorn layer [30] to obtain a double-stochastic affinity matrix $\tilde{\mathbf{M}}_{\text{aff}}$ with maximum k-iteration optimization (k is set 20 enough for optimization). Finally, each positive entry in the affinity matrix $\tilde{\mathbf{M}}_{\text{aff}}$ indicates a matched node pair across two graphs for fine-grained domain adaptation.

Structure-aware Matching Loss. Since graph nodes are drawn from the graphically modeled class-conditional distribution, we align the node pairs across two domains with homogeneous semantics ($\hat{v}^{(\omega)}_s \in \mathcal{G}_s$ and $\hat{v}^{(\omega)}_t \in \mathcal{G}_t$), to adapt the distribution for category ω. Specifically, we propose a structure-aware matching loss to achieve this fine-grained domain adaptation with node-to-node graph matching, which consists of three components as follows,

$$\begin{aligned}\mathcal{L}_{mat} = &\sum_i \frac{1}{\mathcal{N}_s}[\max_j(\tilde{\mathbf{M}}_{\text{aff}}\odot\mathbf{Y}_{\mathbf{\Pi}})_{i,j} - \mathbf{1}]^2\\ &+ \sum_{i,j}\frac{1}{||\mathbf{1}-\mathbf{Y}_{\mathbf{\Pi}}||_1}[\tilde{\mathbf{M}}_{\text{aff}}\odot(\mathbf{1}-\mathbf{Y}_{\mathbf{\Pi}})]^2_{i,j}\\ &+ \sum_{i,j}\frac{1}{\mathcal{N}_s\cdot\mathcal{N}_t}(\mathcal{A}_s\tilde{\mathbf{M}}_{\text{aff}} - \tilde{\mathbf{M}}_{\text{aff}}\mathcal{A}_t)_{i,j},\end{aligned} \tag{5}$$

where the (i,j) entry in $\mathbf{Y}_{\mathbf{\Pi}} \in \mathbb{R}^{\mathcal{N}_s\times\mathcal{N}_t}$ is $\mathbf{1}$ if $v^i_s \in \mathcal{G}_s$ and $v^j_t \in \mathcal{G}_t$ are in the same category ω, otherwise $\mathbf{0}$, and $\tilde{\mathbf{M}}_{\text{aff}} \in \mathbb{R}^{\mathcal{N}_s\times\mathcal{N}_t}$ is the node affinity. The first term works on correctly matched node pairs and enhances the *best-matching* of correct cases, named True-positive Enhancement (TE) (as the *Red* entries of Figure 2 $\hat{\mathbf{Y}}_{\mathbf{\Pi}}$). The second term evaluates the difference between the node affinity and ground-truth to suppress wrongly activated cases, i.e., False-positive Suppression (FS) (as the *Grey* entries of Figure 2 $\hat{\mathbf{Y}}_{\mathbf{\Pi}}$). Besides, we introduce structure-aware Quadratic Constrains (QC) as the third term to minimize the structural difference of matched node pairs in a local

Method	Backbone	person	rider	car	truck	bus	train	motor	bike	mAP	SO/ GAIN
CFFA [42]$_{CVPR'20}$	VGG-16	34.0	46.9	52.1	30.8	43.2	29.9	34.7	37.4	38.6	20.8/ 17.8
EPM [12]$_{ECCV'20}$		41.9	38.7	56.7	22.6	41.5	26.8	24.6	35.5	36.0	18.4/ 17.6
RPNPA [41]$_{CVPR'21}$		33.6	43.8	49.6	32.9	45.5	46.0	35.7	36.8	40.5	20.8/ 19.7
UMT [5]$_{CVPR'21}$		33.0	46.7	48.6	**34.1**	**56.5**	46.8	30.4	37.4	41.7	21.8/ 19.9
MeGA [34]$_{CVPR'21}$		37.7	**49.0**	52.4	25.4	49.2	**46.9**	34.5	39.0	41.8	24.4/ 17.4
ICCR-VDD [36]$_{ICCV'21}$		33.4	44.0	51.7	33.9	52.0	34.7	34.2	36.8	40.0	22.8/ 17.2
KTNet [32]$_{ICCV'21}$		46.4	43.2	60.6	25.8	41.2	40.4	30.7	38.8	40.9	18.4/ 22.5
SSAL [21]$_{NeurIPS'21}$		45.1	47.4	59.4	24.5	50.0	25.7	26.0	38.7	39.6	20.4/ 19.2
SIGMA (ours)		**46.9**	48.4	**63.7**	27.1	50.7	35.9	34.7	**41.4**	**43.5**	18.4/ **25.1**
GPA [37]$_{CVPR'20}$	ResNet-50	32.9	46.7	54.1	24.7	45.7	41.1	32.4	38.7	39.5	22.8/ 16.7
EPM [12]$_{ECCV'20}$		39.9	38.1	57.3	28.7	50.7	37.2	30.2	34.2	39.5	24.2/ 15.3
DIDN [18]$_{ICCV'21}$		38.3	44.4	51.8	28.7	**53.3**	34.7	32.4	40.4	40.5	28.6/ 11.9
DSS [35]$_{CVPR'21}$		42.9	**51.2**	53.6	**33.6**	49.2	18.9	**36.2**	**41.8**	40.9	22.8/ 18.1
SDA [25]$_{ICCV'21}$		38.8	45.9	57.2	29.9	50.2	**51.9**	31.9	40.9	43.3	22.8/ **20.5**
SIGMA (ours)		**44.0**	43.9	**60.3**	31.6	50.4	51.5	31.7	40.6	**44.2**	24.2/ 20.0

Table 1. Results on Cityscapes→Foggy Cityscapes (%) with VGG-16 and ResNet-50 backbone networks. SO represents the source only results and GAIN indicates the adaptation gains compared with the source only model.

neighborhood. Based on the consistent objective of Eq. 1 and Eq. 5 about graph matching, each source node will be aligned to the optimal-matched counterpart in the target domain in the same category, achieving a fine-grained alignment of class-conditional distributions during training.

4.3. Model Optimization

During training, we adopt class-agnostic global alignment [12] on visual features $\{x^i_{s/t}\}^{\mathcal{B}}_{i=1}$ with adversarial loss $\mathcal{L}_{GA}$. Considering the non-grid correspondence among graph nodes and the non-euclidean representation of graphical space [39], we design a Node Discriminator (ND) to align well-matched nodes, consisting a gradient reversed layer [7], three stacked discrimination blocks f_b (each block is FC-LayerNrom-ReLU), and a domain classifier f_{dc} followed with the Binary Cross Entropy (BCE) loss: $\mathcal{L}_{NA} = -\sum_i^{\mathcal{N}_s} \mathcal{D} log\{f_{dc}[f_b(v^i_s)]\} - \sum_i^{\mathcal{N}_t}(1-\mathcal{D}) log\{f_{dc}[f_b(v^i_t)]\}$, where $\mathcal{D}$ is the domain label as [3] and $v^i_{s/t}$ are existing graph nodes. Then, the overall optimization objective of the proposed framework is denoted as:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{node} + \lambda_2 \mathcal{L}_{mat} + \mathcal{L}_{NA} + \mathcal{L}_{GA} + \mathcal{L}_{det}, \quad (6)$$

where $\mathcal{L}_{node}$ is the node classification loss, $\mathcal{L}_{mat}$ is the graph matching loss, $\mathcal{L}_{NA}$ is the node alignment loss, $\mathcal{L}_{GA}$ is the global alignment loss [12] and $\mathcal{L}_{det}$ is the detection loss. $\lambda_{1/2}$ are set 0.1 respectively to control the intensity.

5. Experiments

5.1. Datasets and Evaluation

We conduct extensive experiments on three adaptation scenarios following the standard UDA setting in existing literature [3, 12, 21, 32]. We use the mean Average Precision with different IoU thresholds (mAP$_{IoU}$) for comparison and utilize SO/GAIN to assess the source only results[3] and the adaptation gains compared with the SO. Besides, we also report the results of GA [12] that adopts global alignment [3] on the FCOS [33] detector as our baseline counterpart.

Method	S→C	SO/GAIN	K→C	SO/GAIN
EPM [12]$_{ECCV'20}$	49.0	39.8/ 9.2	43.2	34.4/ 8.8
DSS [35]$_{CVPR'21}$	44.5	34.7/ 9.8	42.7	34.6/ 8.1
MEGA [34]$_{CVPR'21}$	44.8	34.3/ 10.5	43.0	30.2/ **12.8**
RPNPA [41]$_{CVPR'21}$	45.7	34.6/ 11.1	-	-
UMT [5]$_{CVPR'21}$	43.1	34.3/ 8.8	-	-
KTNet [32]$_{ICCV'21}$	50.7	39.8/ 10.9	45.6	34.4/ 11.2
SSAL [21]$_{NeurIPS'21}$	51.8	38.0/ 13.8	45.6	34.9/ 10.7
SIGMA (ours)	**53.7**	39.8/ **13.9**	**45.8**	34.4/ 11.4

Table 2. Comparison results (%) on Sim10K→Cityscapes (S→C) and KITTI→Cityscapes (K→C) with VGG-16 backbone.

Cityscapes→Foggy Cityscapes. The Cityscapes [4] is a street scene datasets captured with on-board cameras under the dry weather condition, which consists of the *train* set (2975 images) and *validation* set (500 images) with eight categories of annotated bounding boxes. Foggy Cityscapes [28] is a synthesized dataset based on the Cityscapes with foggy noise. We explore the weather conditioned domain gap in this adaptation scenario.

Sim10k→Cityscapes. Sim10k [15] is a simulated dataset obtained from the video game Grand Theft Auto V, yielding the domain gap with the real-world scene (Cityscapes). This dataset covers 10,000 images of the annotated bounding boxes in the car category. We perform domain adaptation between synthesized and real-world images and report

[3] Source Only (SO) indicates training with labeled source images and testing on the target data, which is the same as "w/o adapt".

the performance on car category as the common setting.

KITTI→Cityscapes. KITTI [9] is a real-world traffic scene dataset collected from vehicle-mounted cameras, which yields the cross-camera domain gap with Cityscapes (on-board cameras). This dataset covers annotated cars in 7,481 images with cross-camera domain gap for adaptation.

5.2. Implementation Details

We adopt both VGG-16 [29] and ResNet-50 [11] feature extractors, which are implemented with Pytorch [22]. Our model is trained with the Stochastic Gradient Descent (SGD) optimizer with a 0.0025 learning rate, 4 batch-size, momentum of 0.9, and weight decay of 5×10^{-4}. We sample at most 100 graph nodes for each feature map in each domain. Considering the graph matching may fail if no nodes appear in the target domain, we follow [12] to pretrain the framework as a warm-up stage before introducing the BGM adaptor. The adaption-unrelated settings about the object detector strictly follow related works [12, 21, 32].

5.3. Comparison with State-of-the-arts

Cityscapes→Foggy Cityscapes. We present the comparison with VGG-16 and ResNet-50 backbones in Table 1. SIGMA achieves 43.5% and 44.2% mAP, respectively, outperforming existing works by a large margin. Compared with category-level adaptation approaches, e.g., CFFA [42] (38.6%), RPNPA [41] (40.5%), MeGA-CDA [34] (41.8%), KTNet [32] (40.9%), and GPA [37] (39.5%), SIGMA achieves 4.9%, 3.0%, 1.7%, 2.6%, and 4.7% mAP improvements respectively, showing our advantages over existing prototype-based works. Besides, SIGMA surpasses EPM [12], KTNet [32], and SSAL [21] with 7.5%, 2.6%, and 3.9% mAP using the same FCOS [33] object detector.

Sim10k→Cityscapes. The experimental comparison is recorded in the left part of Table 2. SIGMA achieves a 53.7% mAP with the best adaptation gain (13.9% AP), outperforming existing works significantly. Compared with the approaches using the same FCOS [33] object detector, e.g., EPM [12] (49.0% mAP), KTNet [32] (50.7% mAP), SSAL [21] (51.8% mAP), SIGMA gives 4.7%, 3.0%, and 1.9% mAP improvements, verifying our effectiveness.

KITTI→Cityscapes. The comparison results are shown in the right part of Table 2. SIGMA outperforms existing works with a 45.8% mAP and achieves a comparable adaptation gain (11.4% mAP) compared with state-of-the-arts. Compared with EPM [12], KTNet [32] and SSAL [21], our method shows the advantage in terms of adaptation.

5.4. Ablation Studies

We report detailed ablation studies (Table 3) conducted on Cityscapes→Foggy Cityscapes with VGG-16 backbone.

Graph-embedded Semantic Completion. As shown in Table 3, adopting the GSC module can achieve 41.8% mAP with 6.5% mAP gains compared with the GA baseline [12]. We then gradually remove each sub-component to verify its effectiveness. Removing Domain-guided Node Completion (DNC) limits the model optimization under mismatched semantic knowledge (40.4% mAP). Replacing the Graph-guided Memory Bank (GMB) with a common buffer gives 0.8% mAP drops (41.0% mAP) due to the impact of unavoidable noisy samples, and removing Node Discriminator (ND) gives a significant drop (39.4%) due to the severe domain gap in the graphical space.

Bipartite Graph Matching. Introducing the BGM adaptor achieves consistent improvements with a remarkable 43.5% mAP, outperforming the baseline model with 8.2% mAP. Removing Cross Graph Interaction (CGI) gives a 0.7% mAP performance drop (42.8 % mAP) due to the limited interaction between two domains. Replacing the Semantic-aware Node Affinity (SNA) with the simplified strategy in [8] leads to 0.9% mAP drops (42.6% mAP), and remov-

Method	w/o	prsn	rider	car	truc	bus	train	moto	bike	mAP
GA [12]	-	40.3	41.5	54.2	26.7	42.1	15.4	27.1	35.1	35.3
+GSC	DNC	45.2	46.2	57.2	29.1	46.5	31.2	29.2	38.7	40.4
	GMB	43.5	43.8	57.4	**29.4**	48.3	30.4	31.4	41.1	41.0
	ND	44.1	45.2	56.7	28.0	45.9	23.9	32.8	38.7	39.4
	-	45.8	47.6	58.9	27.3	48.6	33.8	32.7	39.3	41.8
+GSC +BGM	CGI	44.4	48.0	58.8	28.4	50.3	**40.5**	31.7	40.8	42.8
	SNA	46.0	46.9	58.8	28.6	48.2	40.4	33.1	39.5	42.6
	SML	46.1	**49.9**	59.1	26.2	**52.5**	27.1	34.6	41.3	42.2
	-	**46.9**	48.4	**63.7**	27.1	50.7	35.9	**34.7**	**41.4**	**43.5**

Table 3. Ablation studies on Cityscapes→Foggy Cityscapes (%).

$\mathcal{N}_s^f$	$\mathcal{N}_t^f$	prsn	rider	car	truc	bus	train	moto	bike	mAP
200	0	41.2	45.1	55.2	26.9	44.2	16.3	28.9	37.0	36.8
0	200	42.4	41.8	55.3	27.7	44.0	21.8	29.2	36.6	37.3
20	20	42.4	44.0	56.5	27.3	45.8	26.6	30.9	38.6	39.0
50	50	44.2	43.4	56.9	**32.2**	45.7	38.6	29.6	37.5	41.0
100	100	**46.9**	48.4	**63.7**	27.1	50.7	35.9	**34.7**	**41.4**	43.5
200	200	44.3	**48.8**	59.0	28.9	51.7	**45.1**	34.2	39.9	**43.9**
500	500	44.4	47.1	58.0	24.4	**52.5**	40.3	31.2	40.1	42.6

Table 4. Results on Cityscapes→Foggy Cityscapes (%) with different node combinations. $\mathcal{N}_{s/t}^f$ represent the maximum sampled nodes from source and target domains in each feature map.

Strategy	Loss	$\text{mAP}_{0.5:0.95}$	$\text{mAP}_{0.5}$	$\text{mAP}_{0.75}$
Single matching	+TE	22.0	42.1	20.3
	+TE+FS	23.8	43.2	23.0
	+TE+FS+QC	**24.0**	**43.5**	**23.5**
Multiple matching	+BCE	23.2	42.9	22.8
	+MSE	23.7	43.1	23.0

Table 5. Results on Cityscapes→Foggy Cityscapes (%) with different matching strategies and loss functions. $\text{mAP}_{0.5:0.95}$ is the averaged mAP from 0.5 to 0.95 IoU with 0.05 intervals. BCE is Binary Cross Entropy and MSE is Mean Squared Error.

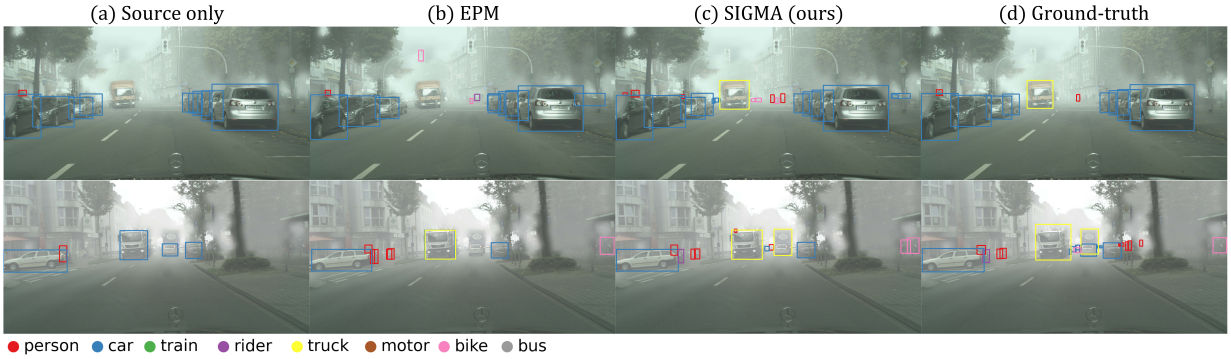

Figure 3. Result comparison on the Cityscapes→Foggy Cityscapes adaptation scenario among (a) the source only model, (b) EPM [12], (c) the proposed SIGMA, and (d) Ground-truth. (Zooming in for best view.)

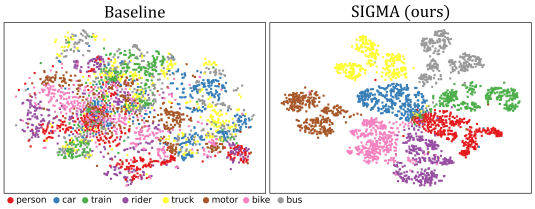

Figure 4. Feature comparison via T-SNE between the baseline model and our method. For each category, we randomly sample object features (marked as squares) inside bounding boxes in the source domain and target domain equally.

ing the Structure-aware Matching Loss (SML) reduces the performance (42.1% mAP). Hence, each sub-component is necessary for SIGMA to achieve state-of-the-art results.

5.5. Sensitivity Analysis

To better understand our method, we investigate the node selection (Table 4) and matching design (Table 5).

Evaluation on the number of nodes. As shown in Table 4, we compare different node combinations ($\mathcal{N}_{s/t}^{f}$ represents the maximum number of nodes sampled from each feature map). Only utilizing source and target nodes (1^{st} and 2^{nd} lines) severely affects the adaptation performance (36.8% and 37.3% mAP) due to deterioration of domain gap in the graphical space. Besides, we find consistent performance improvements from 39.0% to 43.9% (3^{rd} row to 6^{th} row) with the increase of the node number from 20 to 200, because more nodes improve graph matching guided adaptation with better graphical space. However, using too many nodes (e.g., 500) will lead to the difficulty of graph matching optimization with a worse result (42.6% mAP).

Evaluation on matching strategies. We compare different settings between single-matching (each node is matched to the best counterpart) and multiple-matching (each node is matched to all counterparts in the same category) in Table 5. We find single-matching (43.5% $\text{mAP}_{0.5}$) performs relatively better than multiple-matching (43.1% $\text{mAP}_{0.5}$) because singe-matching aligns primary node pairs and relives noisy adaptation on ambiguous nodes. Besides, each component (TE, FS, QC) of the proposed matching loss contributes to the matching-based domain adaptation, yielding consistent $\text{mAP}_{0.5}$ improvements from 42.1% to 43.5%.

5.6. Qualitative Results

Result comparison. We present the comparison among (a) source only, (b) EPM [12], (c) the proposed SIGMA and (d) ground-truth in Figure 3. SIGMA can reduce missing errors, such as the truck in 1^{st} and 2^{nd} lines compared with the category-agnostic method EPM [12]. Besides, our approach also eliminates some classification errors (false-positive cases), such as the rider in 2^{nd} row, showing the advantage in category-level adaptation with well-aligned class-conditional distributions.

Feature comparison. For each category, we randomly sample an equal number of pixels on ResNet-50-based features for each domain (200 pixels/ domain&category) and present the T-SNE comparison with the GA baseline [12] in Figure 4. It can be observed that those similar categories (*person*, *rider*, and *bike*) can be separated clearly on features by our method, which benefits the followed detection head in terms of object recognition significantly.

6. Conclusion

In this paper, we propose a novel framework for DAOD, coined SIGMA. It represents domain information through semantic-complete graphs and model domain adaptation as a graph matching problem, which break the barrier of existing category-level approaches in terms of semantic mismatching and sub-optimal prototype alignment. It adopts a Graph-embedded Semantic Completion module (GSM) to complete mismatched semantics and model class-conditional distributions with graphs. Then, it leverages a Bipartite Graph Matching adaptor (BGM) to achieve fine-grained alignment with a node-to-node matching. Extensive experiments on three benchmarks show that the proposed method outperforms existing approaches significantly.

References

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 4, 5

[2] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021. 2

[3] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *CVPR*, pages 3339–3348, 2018. 1, 2, 4, 6

[4] 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 *CVPR*, pages 3213–3223, 2016. 6

[5] Jinhong Deng, Wen Li, Yuhua Chen, and Lixin Duan. Unbiased mean teacher for cross-domain object detection. In *CVPR*, pages 4091–4101, June 2021. 6

[6] Kexue Fu, Shaolei Liu, Xiaoyuan Luo, and Manning Wang. Robust point cloud registration framework based on deep graph matching. In *CVPR*, pages 8893–8902, 2021. 3, 5

[7] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, pages 1180–1189, 2015. 6

[8] Quankai Gao, Fudong Wang, Nan Xue, Jin-Gang Yu, and Gui-Song Xia. Deep graph matching under quadratic constraint. In *CVPR*, pages 5069–5078, 2021. 3, 5, 7

[9] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, pages 3354–3361, 2012. 7

[10] Jiawei He, Zehao Huang, Naiyan Wang, and Zhaoxiang Zhang. Learnable graph matching: Incorporating graph partitioning with deep feature learning for multiple object tracking. In *CVPR*, pages 5299–5309, 2021. 3, 5

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 7

[12] Cheng-Chun Hsu, Yi-Hsuan Tsai, Yen-Yu Lin, and Ming-Hsuan Yang. Every pixel matters: Center-aware feature alignment for domain adaptive object detector. In *ECCV*, pages 733–748, 2020. 1, 2, 6, 7, 8

[13] Han-Kai Hsu, Chun-Han Yao, Yi-Hsuan Tsai, Wei-Chih Hung, Hung-Yu Tseng, Maneesh Singh, and Ming-Hsuan Yang. Progressive domain adaptation for object detection. In *WACV*, pages 749–757, 2020. 2

[14] Naoto Inoue, Ryosuke Furuta, Toshihiko Yamasaki, and Kiyoharu Aizawa. Cross-domain weakly-supervised object detection through progressive domain adaptation. In *CVPR*, pages 5001–5009, 2018. 2

[15] 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 *ICRA*, pages 746–753, 2017. 6

[16] Taekyung Kim, Minki Jeong, Seunghyeon Kim, Seokeon Choi, and Changick Kim. Diversify and match: A domain adaptive representation learning paradigm for object detection. In *CVPR*, pages 12456–12465, 2019. 2

[17] Congcong Li, Dawei Du, Libo Zhang, Longyin Wen, Tiejian Luo, Yanjun Wu, and Pengfei Zhu. Spatial attention pyramid network for unsupervised domain adaptation. In *ECCV*, pages 481–497. Springer, 2020. 2

[18] Chuang Lin, Zehuan Yuan, Sicheng Zhao, Peize Sun, Changhu Wang, and Jianfei Cai. Domain-invariant disentangled network for generalizable object detection. In *ICCV*, pages 8771–8780, October 2021. 6

[19] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pages 2980–2988, 2017. 4

[20] Eliane Maria Loiola, Nair Maria Maia de Abreu, Paulo Oswaldo Boaventura-Netto, Peter Hahn, and Tania Querido. A survey for the quadratic assignment problem. *Eur. J. Oper. Res.*, 176(2):657–690, 2007. 3

[21] Muhammad Akhtar Munir, Muhammad Haris Khan, M Saquib Sarfraz, and Mohsen Ali. Synergizing between self-training and adversarial learning for domain adaptive object detection. 2021. 2, 6, 7

[22] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, pages 8024–8035, 2019. 7

[23] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 1, 4

[24] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: towards real-time object detection with region proposal networks. In *NeurIPS*, pages 91–99, 2015. 1, 4

[25] Farzaneh Rezaeianaran, Rakshith Shetty, Rahaf Aljundi, Daniel Olmeda Reino, Shanshan Zhang, and Bernt Schiele. Seeking similarities over differences: Similarity-based domain alignment for adaptive object detection. In *ICCV*, pages 9204–9213, 2021. 6

[26] Yu Rong, Wenbing Huang, Tingyang Xu, and Junzhou Huang. Dropedge: Towards deep graph convolutional networks on node classification. *ICLR*, 2020. 4

[27] Kuniaki Saito, Yoshitaka Ushiku, Tatsuya Harada, and Kate Saenko. Strong-weak distribution alignment for adaptive object detection. In *CVPR*, pages 6956–6965, 2019. 1, 2

[28] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Semantic foggy scene understanding with synthetic data. *Int J Comput Vis*, 126(9):973–992, 2018. 6

[29] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 7

[30] Richard Sinkhorn. A relationship between arbitrary positive matrices and doubly stochastic matrices. *Ann. math. stat.*, 35:876–879, 1964. 5

[31] X Yu Stella and Jianbo Shi. Multiclass spectral clustering. In *ICCV*, volume 2, pages 313–313. IEEE Computer Society, 2003. 5

[32] Kun Tian, Chenghao Zhang, Ying Wang, Shiming Xiang, and Chunhong Pan. Knowledge mining and transferring for domain adaptive object detection. In *ICCV*, pages 9133–9142, October 2021. 1, 2, 3, 6, 7

[33] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *ICCV*, pages 9627–9636, 2019. 1, 4, 6, 7

[34] Vibashan VS, Vikram Gupta, Poojan Oza, Vishwanath A. Sindagi, and Vishal M. Patel. Mega-cda: Memory guided attention for category-aware unsupervised domain adaptive object detection. In *CVPR*, pages 4516–4526, June 2021. 5, 6, 7

[35] Yu Wang, Rui Zhang, Shuo Zhang, Miao Li, Yangyang Xia, Xishan Zhang, and Shaoli Liu. Domain-specific suppression for adaptive object detection. In *CVPR*, pages 9603–9612, June 2021. 2, 6

[36] Aming Wu, Rui Liu, Yahong Han, Linchao Zhu, and Yi Yang. Vector-decomposed disentanglement for domain-invariant object detection. *ICCV*, 2021. 6

[37] Minghao Xu, Hang Wang, Bingbing Ni, Qi Tian, and Wenjun Zhang. Cross-domain detection via graph-induced prototype alignment. In *CVPR*, pages 12355–12364, 2020. 1, 2, 3, 4, 6, 7

[38] Junchi Yan, Xu-Cheng Yin, Weiyao Lin, Cheng Deng, Hongyuan Zha, and Xiaokang Yang. A short survey of recent advances in graph matching. In *ACM ICMR*, pages 167–174, 2016. 3

[39] Xu Yang, Cheng Deng, Tongliang Liu, and Dacheng Tao. Heterogeneous graph attention network for unsupervised multiple-target domain adaptation. *IEEE Trans. Pattern Anal. Mach. Intell.*, pages 1–1, 2020. 2, 6

[40] Weilin Zhang and Yu-Xiong Wang. Hallucination improves few-shot object detection. In *CVPR*, pages 13008–13017, 2021. 2

[41] Yixin Zhang, Zilei Wang, and Yushi Mao. Rpn prototype alignment for domain adaptive object detector. In *CVPR*, pages 12425–12434, June 2021. 1, 2, 3, 4, 6, 7

[42] Yangtao Zheng, Di Huang, Songtao Liu, and Yunhong Wang. Cross-domain object detection through coarse-to-fine feature adaptation. In *CVPR*, pages 13766–13775, 2020. 1, 2, 3, 4, 5, 6, 7