Learning under Heterogeneous Shifts: Object-Conditional Meta-Alignment for Domain-Generalized Detection

Anonymous authors
Paper under double-blind review

Abstract

Domain generalization for object detection (DGOD) aims to train object detection models that remain robust under unseen data shifts. Most existing DGOD methods primarily target the single-source setting, where generalization is driven by heavy data augmentation and coarse class-level alignment. However, their performance deteriorates under large, heterogeneous shifts, where distribution changes are localized to object instances and their surrounding context, so class-level alignment fails to address the region-level discrepancies critical for detection. To address this fundamental challenge, we present a novel method for efficiently training the generalized detector under large heterogeneous domain shifts. First, we design a meta-learning based scheme, which efficiently simulates domain shift via episodic meta-train/meta-test splits across source domains, to learn shift-invariant semantic features. Secondly, we introduce an object-centric, likelihood-based similarity for cross-domain samples that underpins a cross-domain feature alignment framework to close the distributional gap induced by heterogeneous domain shifts. The results of comprehensive experiments on a large number of DGOD benchmarks show that our method outperforms previous DGOD methods. Code and prepared data are publicly available on Github (temporally available in supplementary materials).

1 Introduction

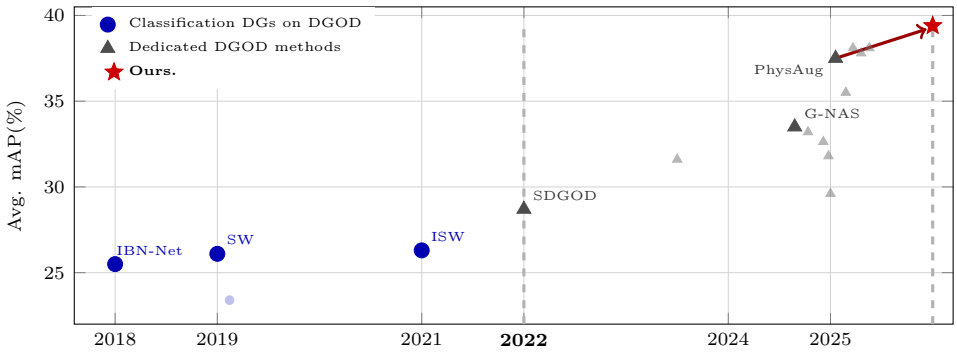


Figure 1: Timeline and performance of DGOD methods. Average mAP (%) versus publication year for representative approaches, highlighting the progression from classification-DG methods adapted to DGOD (blue circles) to existing DGOD methods (gray triangles). Our method (red star) achieves the best average mAP, improving over prior work under large, heterogeneous domain shifts.

Object detection is one of the fundamental tasks in computer vision and has recently seen significant technological breakthroughs Zou et al. (2023). Traditional object detection methods relied on the identically distributed hypothesis, which assumes that the training and testing datasets have identical data distributions Zhu et al. (2021); Tian et al. (2020). However, in real-world applications, object detection models are confronted by substantial challenges from domain shifts, and their performance often suffers significantly

when deployed in unseen domains Chen et al. (2020; 2018). To address these challenges, domain adaptation (DA) Li et al. (2024b); Zhao et al. (2024); Lu et al. (2023); Wu et al. (2022a); Belal et al. (2024); Yao et al. (2021); Wu et al. (2022b) and generalization (DG) Wu & Deng (2022); He et al. (2025); Li et al. (2024a); Liu et al. (2024b); Danish et al. (2024;?); Xu et al. (2023); Wang et al. (2023) methods for object detection have been investigated. In the context of DA methods, it is assumed that the target domain data is available during training. This results in the alignment of source and target data distributions, enabling the model to generalize effectively to the target domain Li et al. (2022b); VS et al. (2023); Zhang & Chou (2024); Hao et al. (2024). The primary limitation of this approach is the need of data collection and model re-training for various target domains, which is time consuming and constrains their practical applications.

Meanwhile, DG goes a step further than DA. The goal of this research line is to enable models trained on existing source domains to perform effectively on unseen domains Zhou et al. (2022). Although DG has been extensively researched in other computer vision tasks such as image classification, it has only recently garnered attention in the area of object detections Lin et al. (2021); Wu & Deng (2022). Unlike classification tasks, which assign each image to a defined set of classes, the domain shift in object detection poses more difficult challenges for training generalized models due to the presence of localizing objects within the image Lin et al. (2021); Wu & Deng (2022). The main research line in this regard is single domain generalization object detection (S-DGOD), which operates under the assumption that only one source domain is available for training generalized models Wu & Deng (2022); He et al. (2025); Li et al. (2024a); Liu et al. (2024b); Danish et al. (2024). S-DGOD methods relies on data augmentation techniques to achieve generalizable results Wu & Deng (2022); He et al. (2025); Li et al. (2024a); Liu et al. (2024b); Danish et al. (2024). However, augmentations mostly alter appearance while preserving object background co-occurrences, so detectors overfit to context that breaks under domain shift. Moreover, augmentations expand the source distribution but do not provably reduce source–target divergence which is the quantity that controls target risk in standard bounds, so the gap to unseen domains remains Ben-David et al. (2010). Motivated by this limitation, several S-DGOD approaches incorporate simple alignment objectives (e.g., KL-divergence regularization, contrastive losses, or ℓ_2 feature matching) adapted from prior classification work to encourage consistent representations between the original and augmented data, thereby reducing the augmentation-induced shift Danish et al. (2024); Xu et al. (2023); Wang et al. (2023) (additional details are provided in the Appendix). In object detection, however, such schemes can be inefficient and less effective because they are often restricted to coarse class-level alignment and overlook the object-level features and instance-level object properties (appearance, scale, context) that vary across objects within the same class.Addressing this challenge will enhance model robustness and reliability in broader real-world applications, particularly by utilizing different source domains for training generalized models, a factor that S-DGOD methods have not taken into account.

To address this fundamental challenge in DGOD, we propose MAML-DGOD, a robust learning framework built on Model-Agnostic Meta-Learning (MAML) that promotes effective alignment across heterogeneous domain distributions. Unlike prior DGOD methods that apply augmentation broadly to the entire training set, we use augmentation during MAML meta-testing to explicitly simulate strong shifts. This design provides a stronger meta-learning signal and encourages the detector to learn representations that better bridge domain gaps. Furthermore, we observe that objects are the primary focus of object detection models. Thus, we propose an alignment algorithm that allows the model to efficiently learn the object-wise semantic consistency of training data from different sources, thereby bridging the domain-shift across different domains.

In summary, our main contributions can be summarized as follows:

- We design a MAML-based training scheme that applies augmentation in the meta-test phase to explicitly simulate *large* domain shifts. This yields a stronger meta-learning signal and encourages the detector to learn shift-invariant semantic representations. To the best of our knowledge, this is the first work to apply MAML to DGOD.

- We propose a novel object-aware conditional density score for efficiently determining the similarity of a sample to the distribution of different domains.

- Based on our proposed similarity score, we design a cross-domain feature alignment framework to encourage the model to learn to fill the gaps between domain shifts via semantic feature representations.

2 Related works

2.1 Domain generalization object detection.

Single source domain generalization object detection. This approach was recently introduced Wu & Deng (2022) with the assumption that there is only one source domain available for training a generalized model capable of performing well on previously unseen domains. An intuitive solution in such a scenario is to propose diverse data augmentation techniques to increase the diversity of a single source data set. These techniques are diverse and include frequency-based transformations Liu et al. (2024b), atmospheric optics theory Xu et al. (2025b), multi-level transformation such as noise, frost for object Lee et al. (2024), apply corruption techniques Danish et al. (2024), or bring spectrum schema Wang et al. (2023) for data augmentation. On the other hand, some methods exploit the power of foundational pretrained large-language models Radford et al. (2021); Li et al. (2024a); Xu et al. (2025a) such as CLIP Radford et al. (2021) or large-vision models He et al. (2025) such as Stable Diffusion Ho et al. (2020) for data augmentation purposes. Beyond augmentation, some studies focus on domain-generalized detection utilizing multi-view learning Xu et al. (2023), distillation He et al. (2025); Wu & Deng (2022), causal learning Rao et al. (2025), or neural network architecture search (NAS) Wu et al. (2024).

However, augmentation-driven training and simple alignment at the domain, image or class level remain too coarse to handle large, heterogeneous domain shifts. In such settings, the mismatch is not uniform across images but varies across localized object instances and their surrounding context. This makes alignment particularly challenging for detection, because spatial localization requires consistency not only in global appearance but also in object-level properties, including object shape, scale, pose, position, and context-dependent cues that can shift differently across domains. In our method, we address the challenge of current approach via object-conditional cross-domain alignment, explicitly reducing inter-domain divergence rather than merely inflating intra-source variance, thereby improving transfer to heterogeneous unseen domains.

2.2 Meta-learning and applications

Meta-learning is regarded as a promising approach for DG Khoee et al. (2024) because it trains models to produce transferable knowledge that can be efficiently utilized for various applications and domains. This is in line with DG's goal of generalizing to unseen target domains. The episodic training scheme is derived from MAML Finn et al. (2017) and has been widely used to address domain generalization for image classification from unknown distributions. The underlying idea is to simulate the distribution gap between train and test in every phase of training by generating episodic train/test splits from the available training data. The MAML application has been widely used for classification tasks Khoee et al. (2024). These promising properties motivate us to employ MAML for building a representation training procedure for DGOD.

3 Method

In this section, we present our approach for learning detection models under heterogeneous source-domain shifts. Our method has two main components: (i) a MAML-DGOD training procedure (together with a domain-splitting strategy and a tailored augmentation schedule) to enhance generalization, and (ii) a cross-domain feature alignment module to promote domain-invariant feature representations. fig. 2 and algorithm 1 illustrate the full pipeline.

3.1 Preliminaries and Notation

Denote source domains as $\mathcal{D}_S = \{D_1, D_2, \ldots, D_N\}$ where $N \geq 1$. Each source domain D_i contains a set of images $\mathcal{X}_i = \left\{\mathbf{x}_m^{(i)}\right\}_{m=1}^{N_i}$ where $\mathbf{x}_m^{(i)} \in \mathbb{R}^{H \times W \times 3}$. Each image $\mathbf{x}$ is annotated with a set of objects

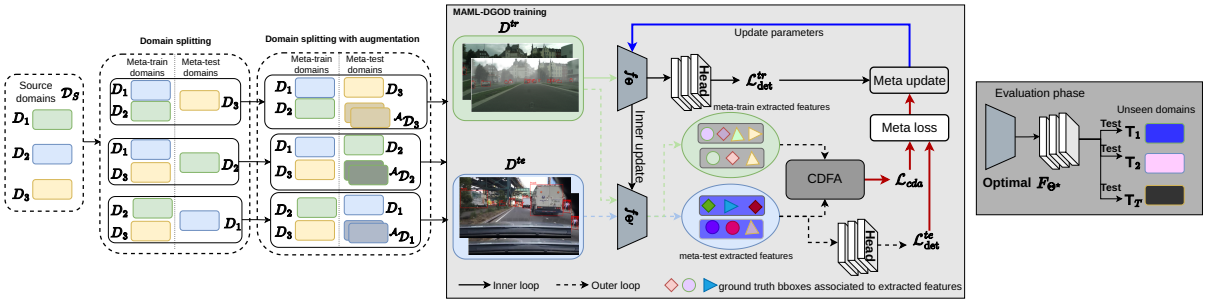


Figure 2: Illustration of our proposed framework. The source domains are partitioned into meta-train and meta-test domains. The meta-test domain is then further diversified through data augmentations. During the training phase, we employ the MAML framework to train the DGOD model on each meta-training and meta-testing domain. Additionally, we encourage the consistency of model learning distribution across different domains through the Cross Domain Feature Alignment (CDFA) module on semantic features from different sources.

$\mathbf{y} = \{(\mathbf{b}_j, c_j)\}_{j=1}^{M(\mathbf{x})}$, where $\mathbf{b}_j \in \mathbb{R}^4$ denotes the bounding box (bbox) parameters and $c_j \in \mathcal{C}$ is the class label from category set $\mathcal{C} = \{1, 2, \ldots, C\}$ and $M(\mathbf{x})$ is the number of annotated objects. Let $a \in \mathcal{A} : (\mathbf{x}, \mathbf{y}) \mapsto (\tilde{\mathbf{x}}, \tilde{\mathbf{y}})$ since a is an augmentation transformation. For each source domain D_i, the augmented domains are denoted as ${}^{\mathcal{A}}\mathcal{D}_i = \bigcup_{a \in \mathcal{A}} a(D_i)$, and ${}^{\mathcal{A}}\mathcal{D}_S = \bigcup_{i=1}^{N} \{{}^{\mathcal{A}}\mathcal{D}_i\}$. We denote unseen target domains $\mathcal{D}_T = \{\mathsf{T}_1, \ldots, \mathsf{T}_T\}$, and denote the detector with its parameters Θ as F_Θ. The detection model relies on a backbone f to extract semantic features for detection further. For an image $\mathbf{x}$ and its annotated bboxes $\mathbf{b}$, the backbone produces $\mathbf{z} = f(\mathbf{x}; \mathbf{b}) \in \mathbb{R}^d$ which is subsequently consumed by the detection heads in F_Θ to yield classification and box regression outputs. The traditional learning objective is to train F_Θ using different source domains and applying a set of augmentations $\mathcal{A}$:

$$\Theta^\star = \arg\min_{\Theta} \; \mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim (\mathcal{D}_S \cup {}^{\mathcal{A}}\mathcal{D}_S)} \; \mathcal{L}_{\text{det}}\big(F_\Theta(\mathbf{x}), \, \mathbf{y}\big) \tag{1}$$

where $\mathcal{L}_{\text{det}}$ is the standard detection loss function.

3.2 MAML-DGOD procedure

We aim to train a detection model that can be applied to different source domains and generalizes to unseen domains. To promote domain generalization, we design an episodic procedure that, at each iteration, partitions the source domains into meta-train and meta-test domains. Moreover, we divide N source domains $\mathcal{D}_S$ into $N-1$ domains for meta-training and one domain for meta-testing (D^{te}). This simulates the domain shift problem that occurs when deploying detection models in real-world scenarios. Assume we have access to a set of data augmentations $\mathcal{A}$. We enrich the meta-test domain by applying these augmentations on it to obtain augmented meta-test domains ${}^{\mathcal{A}}\mathcal{D}_{te} = \bigcup_{a \in \mathcal{A}} a(D^{te})$. We then combine it with the original D^{te} to create a variety of meta-test domains. When only one source domain is available for training, we designate it as a meta-train domain, and any augmented domains become meta-test domains. For each iteration, we sample one domain (D^{tr}) from meta-train domains to use in the MAML-DGOD training phase. To train the model using meta-optimization, we sample batch data B^{tr}, B^{te} from meta-train domain D^{tr}, and from meta-test domains ${}^{\mathcal{A}}\mathcal{D}_{te} \bigcup D^{te}$, respectively. We denote these splitting and sampling processes as $\Pi_{\mathcal{A}}(\mathcal{D}_S)$. Model parameters are then updated using gradients aggregated across the resulting episodic meta-batches. Our model can now be optimized as follows:

$$\Theta^\star = \arg\min_{\Theta} \; \mathbb{E}_{(B^{tr}, \mathsf{B}^{te}) \sim \Pi_{\mathcal{A}}(\mathcal{D}_S)} \Big[\mathcal{L}_{\text{outer}}\big(\mathcal{L}_{\text{inner}}(B^{tr}; \Theta), B^{te}\big) \Big] \tag{2}$$

We utilize $\mathcal{L}_{\text{inner}}$ as $\mathcal{L}_{\text{det}}^{tr}$ which is the detection loss on meta-train domain. $\mathcal{L}_{\text{outer}}$ is the meta-objective function that measures the performance of the model after inner updates on meta-train domain. In our method, we use $\mathcal{L}_{\text{outer}}$ as a combination of $\mathcal{L}_{\text{det}}^{te}, \mathcal{L}_{\text{cda}}$, where $\mathcal{L}_{\text{det}}^{te}$, $\mathcal{L}_{\text{cda}}$ are the detection loss on meta-test domain and the loss function for the alignment scheme, respectively.

3.3 CDFA: Cross domain feature alignment for DGOD

fig. 3 shows the cross-domain feature alignment process on semantic feature learning. Our main idea is to align the distribution of semantic features in the meta-train domain with the meta-test domain to force the model to learn consistency across domain invariants. Here, we take into account that objects are the primary target of detection models; thus, we aim to ensure that model learns semantic features consistently for objects across different domains.

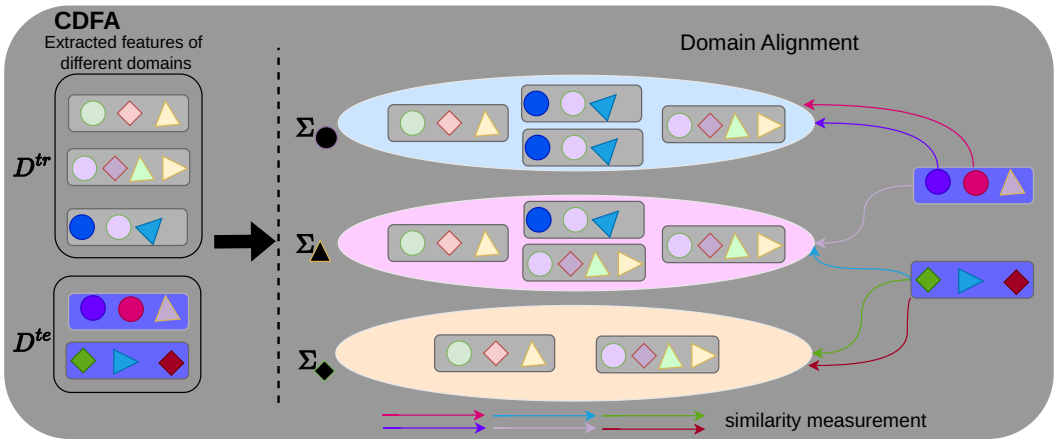


Figure 3: Illustration of CDFA module. Gaussian distributions for each object are obtained based on extracted semantic features of meta-train domain. Cross-domain alignment of meta-train and meta-test was performed for learning invariant representations.

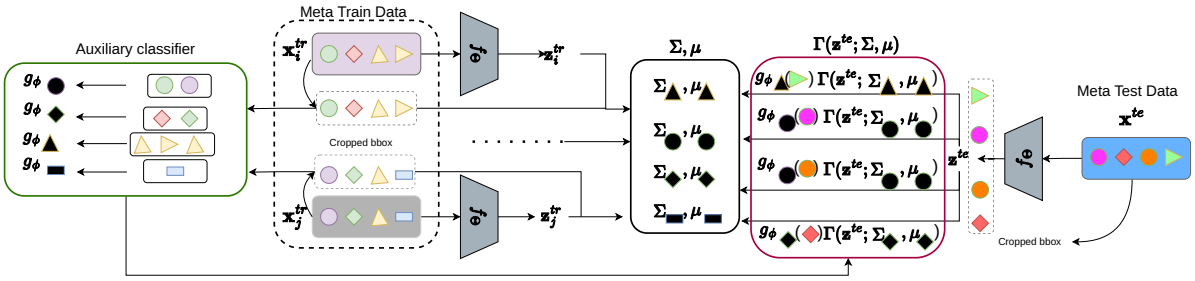


Figure 4: Illustration of a semantic feature's similarity measurement from the meta-test to the meta-train domain. We train auxiliary functions g_ϕ for each object category using cropped ground truth bboxes (gBboxes) from meta-train domain. The gBboxes of meta-test domain will then be inputted to this function to determine its similarity to the training domain. A semantic feature's similarity is determined by how well it matches all distributions of object categories on the meta-train domain.

Therefore, a set of distributions for each object category is obtained using semantic features from the meta-train domain during training. Let $\{\mathbf{z}_i\}$ represents the semantic features extracted by detector's backbone on meta-train domain. For each object category $c \in \mathcal{C}$, we construct a Gaussian probability density function (PDF) represented by $(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$ as below:

$$\boldsymbol{\mu}_c = \frac{1}{n_c}\sum_{i \in \mathcal{S}_c} \mathbf{z}_i, \quad \boldsymbol{\Sigma}_c = \frac{1}{n_c}\sum_{i \in \mathcal{S}_c} (\mathbf{z}_i - \boldsymbol{\mu}_c)(\mathbf{z}_i - \boldsymbol{\mu}_c)^\top \tag{3}$$

where $\mathcal{S}_c$ and n_c represent the set of ground truth bboxes and number of samples for the object class c. Following up Chen et al. (2010); Ledoit & Wolf (2004), we employ standard shrinkage to build robust $\boldsymbol{\Sigma}_c$,

which efficiently reduces large variance and improves numerical stability, and is efficient for further alignment. Resultantly, the variance in eq. (3) becomes $\mathbf{\Sigma}_c \leftarrow (1-\rho)\mathbf{\Sigma}_c + \rho\,\sigma^2 I$ for $\rho \in [0.01, 0.1]$. The distribution $(\boldsymbol{\mu}, \mathbf{\Sigma})$ represents the distribution of all object categories on meta-train domain.

Additionally, we design an alignment procedure that encourages broader object conditional support in the learned semantics, increasing the chance that unseen-domain samples lie in high-density regions of the representation space and thereby improving out-of-distribution generalization. To this end, we develop a likelihood-based score to measure the similarity of a semantic feature to distributions of meta-train domain. Firstly, we define a conditional Gaussian density of a semantic feature $\mathbf{z}_i$ under object category c to $(\boldsymbol{\mu}_c, \mathbf{\Sigma}_c)$, denoted by $\Gamma(\mathbf{z}_i; \boldsymbol{\mu}_c, \mathbf{\Sigma}_c)$ as below:

$$\Gamma\big(\mathbf{z}_i; \boldsymbol{\mu}_c, \mathbf{\Sigma}_c\big) = \frac{\exp\Big(-\frac{1}{2}(\mathbf{z}_i - \boldsymbol{\mu}_c)^\top \mathbf{\Sigma}_c^{-1}(\mathbf{z}_i - \boldsymbol{\mu}_c)\Big)}{\sqrt{(2\pi)^d \det \mathbf{\Sigma}_c}} \tag{4}$$

Consequently, we develop a likelihood-based score that measures the similarity of a feature to distributions $(\boldsymbol{\mu}, \mathbf{\Sigma})$ while taking into account object information that aims to encourage the model's learning of consistency in feature distribution toward objects across various domains. We first measure how similar an annotated object $(\mathbf{b}_i,\, c_i)$ of an image $\mathbf{x}^{te}$ is to the distribution of this object category on meta-train. The process of similarity measurement between a semantic feature of meta-test domain to Gaussian PDFs of meta-train domain is illustrated in fig. 4. We utilize and train an auxiliary function (or posterior) g_ϕ on cropped ground truth bboxes of each object category in the meta-train domain. We use this function to compute a likelihood of a ground truth bbox $\mathbf{b}_i$ from meta-test domain to meta-train domains as follows:

$$q_i(c) \;=\; g_\phi\big(I(\mathbf{b}_i)\big) \in \Delta^{C-1} \tag{5}$$

where $I(\mathbf{b}_i)$ is the image crop and resize operator from the original image using $\mathbf{b}_i$, and Δ is the dimensional vector of object probabilities. Object probability $q_i(c)$ is use to guide the alignment process focusing on the entire information of objects, such as area, texture, shape, and other characteristics. Therefore, we incorporate this probability score with conditional density and employ gated mixture weights Gao et al. (2023) to construct a formulation of the overall similarity of a meta-test semantic feature $\mathbf{z}_i$ to a Gaussian distribution $(\boldsymbol{\mu}_c, \mathbf{\Sigma}_c)$, as below:

$$p^{(i)}\big(\mathbf{z}_i\big) = \sum_{c=1}^{C} \tilde{\pi}_{ic}\, \Gamma\big(\mathbf{z}_i; \boldsymbol{\mu}_c, \mathbf{\Sigma}_c\big) \tag{6}$$

with

$$\tilde{\pi}_{ic} \;\propto\; \pi_c^{tr}\; q_i(c)^\beta, \pi_c^{tr} \;=\; \frac{n_c + \alpha}{\sum_{k=1}^{C}(n_k + \alpha)} \tag{7}$$

where π_c^{tr} is the gated mixture weights, n_k is the number of samples of object category k, and $\alpha \geq 0, \beta > 0$. Finally, we optimize cross-domain feature alignment through alignment energy with the negative log-likelihood of $\mathbf{z}_i$ under gated mixture:

$$\mathcal{L}_{\text{cdfa}} = \frac{1}{N_{\text{gBboxes}}} \sum_{i=1}^{N_{\text{gBboxes}}} \Big[-\log \sum_{c=1}^{C} \tilde{\pi}_{ic}\, \Gamma\big(\mathbf{z}_i; \boldsymbol{\mu}_c, \mathbf{\Sigma}_c\big)\Big] \tag{8}$$

where N_{gBboxes} represents the number of features (equivalent to the number of ground truth bboxes) used in the meta-test batch.

In addition, to improve cross-domain alignment, we design a class-wise batch alignment. To begin, we compute Gaussian PDF for each object on the meta-test domain using auxiliary posteriors and soft weights:

$$\mu_c^{te} = \frac{\sum_i q_i(c)\mathbf{z}_i}{\sum_i q_i(c)}, \Sigma_c^{te} = \frac{\sum_i q_i(c)(\mathbf{z}_i - \mu_c^{te})(\mathbf{z}_i - \mu_c^{te})^\top}{\sum_i q_i(c)} \tag{9}$$

We employ 2-Wasserstein Mallasto & Feragen (2017) to measure the distance between distribution semantic features in meta-train and meta-test domains. As a result, we construct a loss function to minimize this

distance as:

$$\mathcal{L}_{\text{cda}}^{W_2} = \sum_{c=1}^{C} \mathrm{w}_c \, W_2^2\Big(\mathcal{N}(\boldsymbol{\mu}_c^{te}, \boldsymbol{\Sigma}_c^{te}), \mathcal{N}(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)\Big) \tag{10}$$

with $\mathrm{w}_c \propto \sum_i q_i(c)$. In each training step, we formulate the meta loss $\mathcal{L}_{\text{outer}}$ to perform a single meta episode, as shown in the equation below:

$$\mathcal{L}_{\text{outer}} = \mathcal{L}_{\text{det}}^{te} + \mathcal{L}_{\text{cda}} = \mathcal{L}_{\text{det}}^{te} + \lambda_{\text{cdfa}} \cdot \mathcal{L}_{\text{cdfa}} + \lambda_{W_2} \cdot \mathcal{L}_{\text{cda}}^{W_2} \tag{11}$$

with $\lambda_{\text{cdfa}}, \lambda_{W_2}$ are weights controlling the contribution of these losses. Finally, we integrate our proposed

Algorithm 1 MAML-DGOD training on different source domains

Input: Source domains $\mathcal{D}_S$, detector $\mathcal{F}_\Theta$, aux g_ϕ, hyperparameters $\rho, \alpha, \eta_{\text{inner}}, \eta_{\text{outer}}$, $\beta, \lambda_{\text{cdfa}}, \lambda_{W_2}$, and augmentations $\mathcal{A}$

for iteration $= 1, 2, \ldots$ **do**
  Initialize gradient Υ_Θ as 0
  Meta domain splitting:
  for *each* meta-test domain $D^{te} \in \mathcal{D}_S$ **do**
    Build ${}^{\mathcal{A}}\mathcal{D}_{te} \leftarrow \{\, a(D^{te}) : a \in \mathcal{A} \,\}$ (online).
    if $\mathcal{D}_S \setminus \{D^{te}\} = \varnothing$ **then**
      $\mathcal{D}_{\text{mtrain}} \leftarrow \{D^{te}\}$; $\mathcal{D}_{\text{mtest}} \leftarrow {}^{\mathcal{A}}\mathcal{D}_{te}$
    else
      $\mathcal{D}_{\text{mtrain}} \leftarrow \mathcal{D}_S \setminus \{D^{te}\}$
      $\mathcal{D}_{\text{mtest}} \leftarrow \{D^{te}\} \cup {}^{\mathcal{A}}\mathcal{D}_{te}$
    Sample a meta-train domain $D^{tr} \in \mathcal{D}_{\text{mtrain}}$
  MAML-DGOD phase:
    Train g_ϕ on GT bboxes crops from D^{tr}
    Sample support batch B^{tr} from D^{tr}
    Sample meta-test batch B^{te} from $\mathcal{D}_{\text{mtest}}$
  Meta train:
    Compute *inner* loss: $\mathcal{L}_{\text{inner}} \leftarrow \mathcal{L}_{\text{det}}(B^{tr}; \Theta)$
    $\Theta' \leftarrow \Theta - \eta_{\text{inner}} \nabla_{\mathcal{L}_{\text{inner}}}$
  Meta test:
    Compute *outer* loss: $\mathcal{L}_{\text{outer}}$ via equation 11
  Meta optimization:
    $\Upsilon_\Theta \leftarrow \Upsilon_\Theta + \psi \mathcal{L}_{\text{inner}} + (1 - \psi)\mathcal{L}_{\text{outer}}$
  Update model parameters $\Theta \leftarrow \Theta - \eta_{\text{outer}} \, \Upsilon_\Theta$
Return *optimal* $\mathcal{F}_{\Theta^\star}$

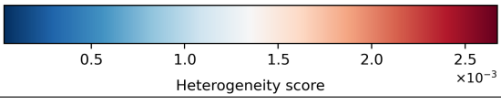
MAML with CDFA to build our MAML-DGOD training procedure as in algorithm 1

4 Experiment

4.1 Settings

Dataset Settings. Similar to many existing DGOD methods Lin et al. (2021); He et al. (2025); Danish et al. (2024); Xu et al. (2025b); Lee et al. (2024); Wu et al. (2024); Vidit et al. (2023); Rao et al. (2025), we employ Diverse Weather Wu & Deng (2022), Real to Artistic Danish et al. (2024) and Corruptions Michaelis et al. (2019) which are popular DGOD benchmarks for evaluating DGOD methods. Additionally, we utilize some public datasets from different real-world scenarios to conduct two additional benchmarks, designated as Cross Maritime Iancu et al. (2021); Shao et al. (2018); Yang et al. (2024) and Cross Locale Arya et al. (2024), resulting in a total of five DGOD benchmarks. We summarize datasets for all benchmarks in table 1. For Diverse Weather, Corruptions, and Real to Artistic benchmarks, we utilize only one dataset for the source

Table 1: Benchmark datasets for DGOD utilized in our experiments. Final column presents the heterogeneity level calculated by KID Bińkowski et al. (2018) score of unseen domain in relation to source domains. A greater heterogeneity score reflects more difference between source and unseen domains, resulting in greater difficulty for detection models. ⊘: We omit heterogeneity computation for datasets that are not used in the evaluation.



Benchmark	Datasets	Abbrev.	Train	Val	Test	Object category	Heterogeneity
Diverse Weathers Hassaballah et al. (2021); Cordts et al. (2016); Yu et al. (2020)	Daytime Clear	DC	19395	8313	–	bike,bus,car,motor,person,rider,truck	⊘
	Daytime Foggy	DF	–	–	3775	bike,bus,car,motor,person,rider,truck	
	Dusk Rainy	DR	–	–	3501	bike,bus,car,motor,person,rider,truck	
	Night Sunny	NS	–	–	26158	bike,bus,car,motor,person,rider,truck	
	Night Rainy	NR	–	–	2494	bike,bus,car,motor,person,rider,truck	
Cross Maritime	AboShips Iancu et al. (2021)	SA	3010	1500	2369	ship	
	SeaShips Shao et al. (2018)	SS	1750	1750	3500	ship	
	Canal Yang et al. (2024); Jeong et al. (2025)	SC	750	391	321	ship	
Real to Artistic.	PASCAL VOC	VOC	16551	4992	–	horse,motorbike,car,dog,sheep,pottedplant,train, sofa,person,bicycle,tvmonitor,diningtable,bottle, cat,aeroplane,bird,chair,bus,boat,cow	⊘
	Clipart	clipart	–	–	1000	*same as PASCAL VOC*	
	Comic	comic	–	–	2000	cat,car,dog,bird,person,bicycle	
	Watercolor	watercolor	–	–	2000	cat,car,dog,bird,person,bicycle	
Cross Locale Arya et al. (2024)	China MotorBike	Cn	1186	198	593	D00, D10, D20, D40	
	Czech	Cz	1697	283	849	D00, D10, D20, D40	
	India	In	4624	770	2312	D00, D10, D20, D40	
	Japan	Jp	6304	1050	3152	D00, D10, D20, D40	
	Norway	No	4897	816	2448	D00, D10, D20, D40	
	United States	USA	2883	480	1442	D00, D10, D20, D40	
Corruptions	CityScapes	C	2975	500	–	person,bike,motor,bus,truck,train,rider,car	⊘
	Brightness	Bright	–	–	500	person,bike,motor,bus,truck,train,rider,car	
	Contrast	Contrast	–	–	500	person,bike,motor,bus,truck,train,rider,car	
	Defocus Blur	Defocus	–	–	500	person,bike,motor,bus,truck,train,rider,car	
	Elastic Transforms	Elastic	–	–	500	person,bike,motor,bus,truck,train,rider,car	
	Fog	Fog	–	–	500	person,bike,motor,bus,truck,train,rider,car	
	Frost	Frost	–	–	500	person,bike,motor,bus,truck,train,rider,car	
	Gaussian Noise	Gauss.	–	–	500	person,bike,motor,bus,truck,train,rider,car	
	Glass Blur	Glass	–	–	500	person,bike,motor,bus,truck,train,rider,car	
	Impulse Noise	Impulse	–	–	500	person,bike,motor,bus,truck,train,rider,car	
	JPEG Compression	JPEG	–	–	500	person,bike,motor,bus,truck,train,rider,car	
	Motion Blur	Motion	–	–	500	person,bike,motor,bus,truck,train,rider,car	
	Pixelate	Pixel	–	–	500	person,bike,motor,bus,truck,train,rider,car	
	Shot Noise	Shot	–	–	500	person,bike,motor,bus,truck,train,rider,car	
	Snow	Snow	–	–	500	person,bike,motor,bus,truck,train,rider,car	
	Zoom Blur	Zoom	–	–	500	person,bike,motor,bus,truck,train,rider,car	

domain $\mathcal{D}_S$, while the other datasets serve as the unseen target domains $\mathcal{D}_T$. For the remaining benchmarks, we employ one dataset as the unseen target domain $\mathcal{D}_T$, while the rests serve as different source domains $\mathcal{D}_S$. We define a pattern "D_1&D_2&...&D_N to $\mathsf{T}_1, \mathsf{T}_2, ..., \mathsf{T}_T$" in the subsequent experimental results associated with our data setting. For example, "SA&SS to SC" indicates model trains on source domains SA and SS, while only being evaluated on unseen domain SC.

Experiment Settings. We adopt the same experimental setup as many existing DGOD methods Wu & Deng (2022); Danish et al. (2024); Xu et al. (2025b); Lee et al. (2024); Rao et al. (2025). Specifically, we utilize Faster R-CNN Ren et al. (2015) with ResNet101 He et al. (2016) backbones pretrained on ImageNet Deng et al. (2009) as our baseline detector. We utilize defocus Abuolaim et al. (2021); Rajagopalan et al. (2023), albumentation Buslaev et al. (2020), color jittering, contrast adjustment, sharpening, rotation, shearing, and translation Li et al. (2022d) as augmentations $\mathcal{A}$.

We employ ResNet18 pretrained on ImageNet as a classifier for auxiliary function g_ϕ. We implement $I(\mathbf{b}_i)$ to crop and resize ground truth ROIs to size 112×112. We train auxiliary model models for 200 iterations each episode. We set $\lambda_{\text{cdfa}}, \lambda_{W_2}, \alpha, \beta, \rho$ to $3\text{e}-4$, $1\text{e}-4$, 20, 0.5 and 0.05 respectively. Our model is trained for 100k iterations using a batch size of 16, a learning rate of 0.002 for both inner and outer update, and SGD optimizer. Our experiments are conducted on NVIDIA H200 equipped with 140GB GPU memory. Our implementation is based on MMDetection library. We report mean Average Precision (mAP50) across object categories.

4.2 Results and Comparisons

Performance on DGOD benchmarks. We benchmark our method against prior DGOD approaches under a fully unseen-target evaluation protocol. For Cross Maritime and Cross Locale benchmarks, we compare our method against only reproducible DGOD methods. In all tables, bold and underline denote

the best and second-best results among DGOD methods; '-' indicates the lack of a performance report; and ▷ indicates the results are obtained directly from referred works.

Table 2: Performance of DGOD methods on Diverse Weather benchmark.

Methods	Daytime Clear	Night Sunny	Daytime Foggy	Dusk Rainy	Night Rainy	Average
Vanilla Faster-RCNN Ren et al. (2015)	48.1	34.4	32	26	12.4	26.2
Classification based DG methods for DGOD						
▷SW Pan et al. (2019b) *(ICCV'19)*	50.6	33.4	30.8	26.3	13.7	26.1
▷IBN-Net Pan et al. (2019a) *(ICCV'19)*	49.7	32.1	29.6	26.1	14.3	25.5
▷IterNorm Huang et al. (2019) *(CVPR'19)*	43.9	29.6	28.4	22.8	12.6	23.4
MASF Huang et al. (2019) *(NeurIPS'19)*	52.4	35.6	31.2	26.3	14.6	26.9
▷ISW Choi et al. (2021) *(CVPR'21)*	51.3	33.2	31.8	25.9	14.1	26.3
▷SHADE Zhao et al. (2022) *(ECCV'22)*	-	33.9	33.4	29.5	16.8	28.4
Vision-only DGOD methods						
▷SDGOD Wu & Deng (2022) *(CVPR'22)*	56.1	36.6	33.5	28.2	16.6	28.7
▷SRCD Rao et al. (2025) *(TNNLS'25)*	-	36.7	35.9	28.8	17	29.6
▷OA-Mix Lee et al. (2024) *(AAAI'24)*	56.4	38.6	38.1	33.8	14.8	31.3
▷OADG Lee et al. (2024) *(AAAI'24)*	55.8	38	38.3	33.9	16.8	31.8
▷G-NAS Wu et al. (2024) *(AAAI'24)*	58.4	45	36.4	35.1	17.4	33.5
▷UFR Liu et al. (2024b) *(CVPR'24)*	58.6	40.8	39.6	33.2	19.2	33.2
▷Div Danish et al. (2024) *(CVPR'24)*	50.6	39.4	35.6	37	22	33.5
▷DivAlign Danish et al. (2024) *(CVPR'24)*	52.8	42.5	37.2	38.1	24.1	35.5
▷PhysAug Xu et al. (2025b) *(AAAI'25)*	60.2	44.9	40.8	41.2	23.1	37.5
Ours.	60.2	45	41.9	**41.5**	**25.1**	**38.4**
LLMs/VLMs based DGOD methods						
▷CLIP Gap Vidit et al. (2023) *(CVPR'23)*	51.3	36.9	38.5	32.3	18.7	31.6
▷PDOC Li et al. (2024a) *(CVPR'24)*	53.6	38.5	39.1	33.7	19.2	32.6
▷VLTDet Hümmer et al. (2024) *(ACCV'24)*	60.5	44.6	42.3	38.4	22.1	36.9
▷VLM-DGOD Xu et al. (2025a) *(MM'25)*	59.8	45	42.1	41.3	23.8	38.1
▷Diff. Guided (SD-2.1) He et al. (2025) *(CVPR'25)*	-	**49.3**	**44.7**	37.1	20	37.8
▷Diff. Guided (SD-1.5) He et al. (2025) *(CVPR'25)*	-	48.7	**44.7**	37.4	21.7	38.1

Our results in tables 2 to 5 show that our method delivers consistently strong performance across DGOD benchmarks, achieving state-of-the-art accuracy in multiple settings and, in table 3, even outperforming several domain adaptation baselines that leverage target-domain data. The proposed MAML training, together with effective feature alignment, improves generalization when the source–target gap is moderate. Notably, the gains are larger under severe shifts, most clearly on the Corruptions and Real-to-Artistic evaluations reported in tables 3 and 4.

Looking deeper these tables, we observe that augmentation-only methods (e.g., OA-Mix, Div, PhysAug) or simple class-based alignment (e.g., SDGOD, SRCD, DivAlign, OADG) that globally align source distributions remain insufficient under the larger shifts. In contrast, LLM/VLMs based DGOD methods such as *Diff. Guided* He et al. (2025) and VLM-DGOD Xu et al. (2025a) are generally closer to our performance than vision-only methods; however, they rely on large diffusion and language foundation models (Stable Diffusion v1.5/2.1 and CLIP) with billions of parameters, incurring substantially higher compute and memory costs (this limitation was stated in their papers He et al. (2025); Xu et al. (2025a)). It once again demonstrates the efficiency and robustness of our method for training a generalized detection model using different source datasets.

Table 3: Performance of DGOD models on unseen datasets of Real to Artistic benchmark. ◇ indicates the results are obtained directly from He et al. (2025).

Method	VOC	Clipart	Comic	Watercolor	Avg.
◇Vanilla Faster-RCNN Ren et al. (2015)	81.8	25.7	18.9	44.5	29.7
Domain adaptation methods					
◇SWDA Saito et al. (2019) *(CVPR'19)*	-	-	29.4	53.3	-
◇DBGL Chen et al. (2021) *(ICCV'21)*	-	-	29.7	53.8	-
◇AT Li et al. (2022c) *(CVPR'22)*	-	49.3	-	59.9	-
◇D-ADAPT Jiang et al. (2022) *(ICLR'22)*	-	49.0	40.5	-	-
◇TIA Zhao & Wang (2022) *(CVPR'22)*	-	46.3	-	-	-
◇LODS Li et al. (2022a) *(CVPR'22)*	-	45.2	-	58.2	-
◇CIGAR Liu et al. (2023) *(CVPR'23)*	-	46.2	-	-	-
◇CMT Cao et al. (2023) *(CVPR'23)*	-	47	-	-	-
◇UMGA Zhang et al. (2024) *(TPAMI'24)*	-	49.9	-	62.1	-
◇CAT Kennerley et al. (2024) *(CVPR'24)*	-	49.1	-	-	-
Vision-only DGOD methods					
◇Div Danish et al. (2024) *(CVPR'24)*	82.1	33.7	25.5	52.5	37.2
◇DivAlign Danish et al. (2024) *(CVPR'24)*	80.1	38.9	33.2	**57.5**	43.2
PhysAug Xu et al. (2025b) *(AAAI'25)*	82.0	32.3	22.9	47.6	34.3
Ours.	81.9	40.4	**34.9**	56.9	**44.1**
LLM/VLMs based DGOD methods					
◇Diff. Guided (SD-2.1) He et al. (2025) *(CVPR'25)*	-	**40.8**	29.7	54.2	41.6
◇Diff. Guided (SD-1.5) He et al. (2025) *(CVPR'25)*	-	32.7	24.9	50.6	36.1

Heterogeneous domain-shift robustness analysis. We plot mAP50 against the Kernel Inception Distance (KID) between domains in fig. 8, where a larger KID indicates a more heterogeneous shift. While prior DGOD baselines show a sharp collapse as KID grows, our method degrades more gradually and consistently achieves the highest mAP50 at large KID values. This trend supports that our object-centric alignment

Table 4: Performance of DGOD models on Corruptions benchmark. We utilize ResNet50 (R50) as the backbone for only this benchmark, similar to the settings of compared methods.

Methods	Clean	Noise			Blur				Weather			Digital					Avg.
		Gauss.	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Contrast	Elastic	Pixel	JPEG	
Vanilla Faster-RCNN Ren et al. (2015)	42.2	0.5	1.1	1.1	17.2	16.5	18.3	2.1	2.2	12.3	29.8	32	24.1	40.1	18.7	15.1	15.4
Vision-only DGOD methods																	
▷FSCE Sun et al. (2021)*(CVPR'21)*	43.1	7.4	10.2	8.2	23.3	20.3	21.5	4.8	5.6	23.6	37.1	38.0	31.9	**40.2**	20.4	23.2	21.0
▷OA-Mix Lee et al. (2024)*(AAAI'24)*	42.7	7.2	9.6	7.7	22.8	18.8	21.9	5.4	5.2	23.6	37.3	38.7	31.9	**40.2**	20.2	22.2	20.8
▷OADG Lee et al. (2024)*(AAAI'24)*	43.4	8.2	10.6	8.4	24.6	20.5	22.3	4.8	6.1	**25.0**	38.4	39.7	32.8	**40.2**	22.0	23.8	21.8
▷PhysAug Xu et al. (2025b)*(AAAI'25)*	42.6	14.3	17.0	11.9	25.6	19.1	25.5	3.9	8.6	21.3	35.3	39.5	27.5	39.1	28.9	19.9	22.6
Ours.*(R50)*	42.4	**21.68**	**24.3**	**21.42**	**29.6**	**24.14**	**29.12**	**6.82**	**19.7**	**25.02**	36.6	40	32.74	39.64	**30.06**	**29.44**	**27.35**
LLM/VLMs based DGOD methods																	
▷Diff. Guided (SD-2.1) He et al. (2025)*(CVPR'25)*	42.2	8.2	10.5	8.2	21.6	12.4	20.1	3.0	3.1	24.5	39.2	39.5	35.8	38.7	23.1	19.6	20.5
▷Diff. Guided (SD-1.5) He et al. (2025)*(CVPR'25)*	42.1	11.0	13.6	10.8	25.0	14.2	21.4	3.4	5.4	24.0	**39.6**	**40.3**	**36.3**	39.2	18.9	16.0	21.3
▷VLM-DGOD Xu et al. (2025a)*(MM'25)*	42.4	17.4	20.5	15.6	26.6	19.7	25.3	4.4	10.0	23.3	39.3	40.0	31.9	39.4	28.7	21.2	24.2

Table 5: Performance of DGOD models on Cross Locale and Maritime benchmarks.

Methods	Cross Locale							Cross Maritime			
	Cn&Cz&In&Jp&No to USA	Cn&Cz&In&Jp&USA to No	Cn&Cz&In&No&USA to Jp	Cn&Cz&Jp&No&USA to In	Cn&In&Jp&No&USA to Cz	USA&Cz&In&Jp&No to Cn	Avg.	SA&SS2SC	SA&SC2SS	SC&SS2SA	Avg.
Vanilla Faster RCNN Ren et al. (2015)	30.18	7.75	16.9	8.03	11.48	26.33	16.78	38.4	36.5	10.5	37.77
CLIP Gap Vidit et al. (2023)*(CVPR'23)*	33.43	8.68	20	9.05	13.25	27.45	18.64	45.93	36.11	11.61	42.66
OA-Mix Lee et al. (2024)*(AAAI'24)*	35.43	8.93	22.08	10.3	14.73	29.13	20.1	46.82	41.28	11.7	44.97
G-NAS Wu et al. (2024)*(AAAI'24)*	36.45	9.33	22.63	10.53	15.15	29.78	20.65	47.4	43.12	11.9	45.97
DivAlign Danish et al. (2024)*(CVPR'24)*	37.05	10.88	25.2	12.75	15.98	30.38	22.04	49.6	54.8	12.3	51.33
PhysAug Xu et al. (2025b)*(AAAI'25)*	38.45	10.75	28.78	13.83	17.7	31.75	23.54	54.4	50.2	14.7	53
Ours.	**41.65**	**17.9**	**29.78**	**17.85**	**20.53**	**34.93**	**27.11**	**55.7**	**59**	**19.3**	**56.8**

and episodic meta-learning better handle ROI-level discrepancies that dominate under large heterogeneous shifts.

4.3 Ablation studies

During development of our method, we utilize Cross Maritime benchmark to perform ablation studies and find the best hyperparameters before using it as one default setting across all experiments.

Ablation on loss components. table 6 shows that each component significantly improves overall performance. Our proposed CDFA serves a more important role here, as performance suffers significantly in the absence of this loss component. fig. 6 shows the ablative study of the contribution ratio of meta-training and meta-testing losses in our method, which is controlled by the hyperparameter ψ in algorithm 1. A value near 0.8 yields optimal results. Additional ablations for other hyperparameters are provided in the Supplementary Materials. Overall, these studies suggest that, for a new application, it is sufficient to tune only $\psi \in [0.1, 0.9]$ and $\lambda_{\text{cda}} \in [10^{-4}, 10^{-3}]$, while keeping the remaining hyperparameters fixed.

Table 8: Integration of our training scheme with other detectors.

Detector (backbone)	SA&SS2SC	SA&SC2SS	SC&SS2SA
YOLOX-S (CSPDarknet-53)Ge et al. (2021)	32.13	30.8	8.2
+**Ours.**	44.2	52.9	15.1
DETR (R50)Carion et al. (2020)	35.9	32.4	8.6
+**Ours.**	50.1	54.8	15.2
Grounding DINO (SWIN-T)Liu et al. (2024a)	48.3	55.16	16.8
+**Ours.**	60.5	64.2	21.9

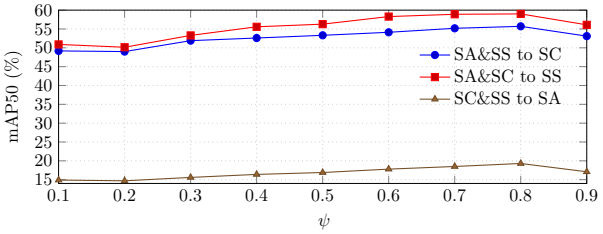
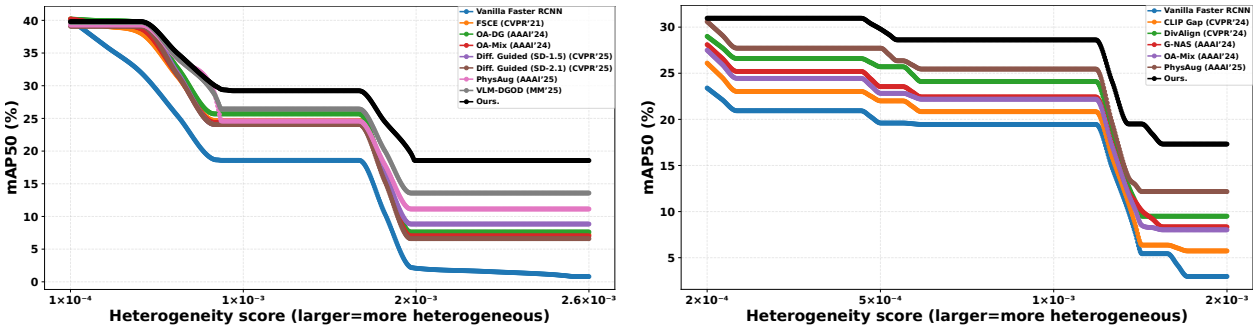


Figure 6: Sensitivity to hyperparameter ψ balancing meta-train and meta-test losses.



(a) Corruptions.

(b) Cross Locale.

Figure 5: Performance-heterogeneity curves on Corruptions and Cross Locale benchmarks.

Table 6: The effects of different loss components to the performance of our method.

Method	$\mathcal{L}_{\text{det}}^{te}$	$\mathcal{L}_{\text{cdfa}}$	$\mathcal{L}_{\text{cda}}^{W_2}$	SA&SS to SC	SA&SC to SS	SC&SS to SA
Faster-RCNN				38.4	36.5	10.5
Ours.	✓			49.2	48.8	14.3
	✓	✓		54.5	56.2	18.7
	✓		✓	51.3	52.6	16.9
	✓	✓	✓	**55.7**	**59**	**19.3**

Table 7: Integration of our MAML with different alignment schemes. KL: Kullback-Leibler divergence.

Alignment Scheme	SA&SS2SC	SA&SC2SS	SC&SS2SA
Our MAML+(KL+L2-distance Danish et al. (2024))	50.5	52.3	14.9
Our MAML+(KL+Contrastive learning Liu et al. (2024b))	51.4	49.4	15.3
Our MAML+(KL+Contrastive learning Lee et al. (2024))	49.2	51.5	15.1
Our MAML+Our CDFA (Ours.)	**55.7**	**59**	**19.3**

Integration of our MAML with other alignment schemes. table 7 show the appliaapplicability of our designed MAML scheme on integrating with existing alignment schemes using in DGOD, including KL+L2-distance Danish et al. (2024), cosine similarity Vidit et al. (2023), and KL+Contrastive learning Lee et al. (2024). This results also highlight that our CDFA outperforms these alternative alignment schemes when integrated with our MAML-based training scheme. This demonstrates the effectiveness of our proposed CDFA in enhancing the generalization performance of detection models across heterogeneous source domains.

Integration of our method with other detectors. table 8 demonstrates that our training scheme is plug-and-play and consistently improves a diverse set of detectors (e.g., YOLOX, DETR, and Grounding DINO). Note that we use Faster R-CNN in the main DGOD comparison since it is the main method in prior DGOD benchmarks, ensuring fair comparison with existing methods.

Comparison with existing MAML for DG. As MAML have yet existed in DGOD, we compare our method with MASF Dou et al. (2019), which is a MAML-based domain generalization method for classification. Results in table 2 show that MASF performs much worse than our method, which indicates that directly applying classification based MAML training scheme for DGOD is not effective.

5 Conclusion

This paper addresses a fundamental challenge in domain generalization for object detection, wherein detection models struggle across heterogeneous data source domains. We employ MAML to train detection models across different source domains to improve the model's generalizability. Additionally, we introduce CDFA, an efficient cross-domain feature alignment framework by establishing a likelihood-based similarity measurement to determine the similarity of semantic features from the distribution of different source domains. We evaluate our method using five challenging benchmarks for DGOD with a substantial quantity of extensive datasets. Comprehensive experiment results demonstrate significant improvements over existing DGOD methods and show the efficiency and robustness of our method across different scenarios.

Acknowledgments

References

Abdullah Abuolaim, Mauricio Delbracio, Damien Kelly, Michael S Brown, and Peyman Milanfar. Learning to reduce defocus blur by realistically modeling dual-pixel data. In *ICCV*, pp. 2289–2298, 2021.

Deeksha Arya, Hiroya Maeda, Sanjay Kumar Ghosh, Durga Toshniwal, and Yoshihide Sekimoto. Rdd2022: A multi-national image dataset for automatic road damage detection. *Geoscience Data Journal*, 11(4): 846–862, 2024.

Atif Belal, Akhil Meethal, Francisco Perdigon Romero, Marco Pedersoli, and Eric Granger. Multi-source domain adaptation for object detection with prototype-based mean teacher. In *WACV*, pp. 1277–1286, 2024.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *ICLR*, 2018.

Alexander Buslaev, Vladimir I. Iglovikov, Eugene Khvedchenya, Alex Parinov, Mikhail Druzhinin, and Alexandr A. Kalinin. Albumentations: Fast and flexible image augmentations. *Information*, 11(2), 2020. ISSN 2078-2489. doi: 10.3390/info11020125.

Shengcao Cao, Dhiraj Joshi, Liang-Yan Gui, and Yu-Xiong Wang. Contrastive mean teacher for domain adaptive object detectors. In *CVPR*, pp. 23839–23848, 2023.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, pp. 213–229. Springer, 2020.

Chaoqi Chen, Zebiao Zheng, Xinghao Ding, Yue Huang, and Qi Dou. Harmonizing transferability and discriminability for adapting object detectors. In *CVPR*, pp. 8869–8878, 2020.

Chaoqi Chen, Jiongcheng Li, Zebiao Zheng, Yue Huang, Xinghao Ding, and Yizhou Yu. Dual bipartite graph learning: A general approach for domain adaptive object detection. In *CVPR*, pp. 2703–2712, 2021.

Yilun Chen, Ami Wiesel, Yonina C Eldar, and Alfred O Hero. Shrinkage algorithms for mmse covariance estimation. *IEEE transactions on signal processing*, 58(10):5016–5029, 2010.

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*, pp. 3339–3348, 2018.

Sungha Choi, Sanghun Jung, Huiwon Yun, Joanne T Kim, Seungryong Kim, and Jaegul Choo. Robustnet: Improving domain generalization in urban-scene segmentation via instance selective whitening. In *CVPR*, pp. 11580–11590, 2021.

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*, June 2016.

Muhammad Sohail Danish, Muhammad Haris Khan, Muhammad Akhtar Munir, M Saquib Sarfraz, and Mohsen Ali. Improving single domain-generalized object detection: A focus on diversification and alignment. In *CVPR*, pp. 17732–17742, 2024.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pp. 248–255, 2009.

Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. *NeurIPS*, 32, 2019.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, pp. 1126–1135, 2017.

Richard Gao, Michael Deistler, and Jakob H Macke. Generalized bayesian inference for scientific simulators via amortized cost estimation. *NeurIPS*, 36:80191–80219, 2023.

Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021.

Yan Hao, Florent Forest, and Olga Fink. Simplifying source-free domain adaptation for object detection: Effective self-training strategies and performance insights. In *ECCV*, pp. 196–213. Springer, 2024.

M. Hassaballah, Mourad A. Kenk, Khan Muhammad, and Shervin Minaee. Vehicle detection and tracking in adverse weather using a deep learning framework. *IEEE Transactions on Intelligent Transportation Systems*, 22(7):4230–4242, 2021.

Boyong He, Yuxiang Ji, Qianwen Ye, Zhuoyue Tan, and Liaoni Wu. Generalized diffusion detector: Mining robust features from diffusion models for domain-generalized detection. In *CVPR*, pp. 9921–9932, 2025.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, June 2016.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 33:6840–6851, 2020.

Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. Fsdr: Frequency space domain randomization for domain generalization. In *CVPR*, pp. 6891–6902, 2021.

Lei Huang, Yi Zhou, Fan Zhu, Li Liu, and Ling Shao. Iterative normalization: Beyond standardization towards efficient whitening. In *CVPR*, pp. 4874–4883, 2019.

Christoph Hümmer, Manuel Schwonberg, Liangwei Zhou, Hu Cao, Alois Knoll, and Hanno Gottschalk. Strong but simple: A baseline for domain generalized dense perception by clip-based transfer learning. In *ACCV*, pp. 4223–4244, 2024.

Bogdan Iancu, Valentin Soloviev, Luca Zelioli, and Johan Lilius. Aboships—an inshore and offshore maritime vessel detection dataset with precise annotations. *Remote Sensing*, 13(5):988, 2021.

Mingi Jeong, Arihant Chadda, Ziang Ren, Luyang Zhao, Haowen Liu, Aiwei Zhang, Yitao Jiang, Sabriel Achong, Samuel Lensgraf, Monika Roznere, et al. Seepersea: Multi-modal perception dataset of in-water objects for autonomous surface vehicles. *IEEE Transactions on Field Robotics*, 2025.

Junguang Jiang, Baixu Chen, Jianmin Wang, and Mingsheng Long. Decoupled adaptation for cross-domain object detection. In *ICLR*, 2022.

Mikhail Kennerley, Jian-Gang Wang, Bharadwaj Veeravalli, and Robby T Tan. Cat: Exploiting inter-class dynamics for domain adaptive object detection. In *CVPR*, pp. 16541–16550, 2024.

Arsham Gholamzadeh Khoee, Yinan Yu, and Robert Feldt. Domain generalization through meta-learning: a survey. *Artificial Intelligence Review*, 57(10):285, 2024.

Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of multivariate analysis*, 88(2):365–411, 2004.

Wooju Lee, Dasol Hong, Hyungtae Lim, and Hyun Myung. Object-aware domain generalization for object detection. In *AAAI*, volume 38, pp. 2947–2955, 2024.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018a.

Deng Li, Aming Wu, Yaowei Wang, and Yahong Han. Prompt-driven dynamic object-centric learning for single domain generalization. In *CVPR*, pp. 17606–17615, 2024a.

Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. Domain generalization with adversarial feature learning. In *CVPR*, pp. 5400–5409, 2018b.

Shuaifeng Li, Mao Ye, Xiatian Zhu, Lihua Zhou, and Lin Xiong. Source-free object detection by learning to overlook domain style. In *CVPR*, pp. 8014–8023, 2022a.

Shuaifeng Li, Mao Ye, Xiatian Zhu, Lihua Zhou, and Lin Xiong. Source-free object detection by learning to overlook domain style. In *CVPR*, pp. 8014–8023, 2022b.

Shuaifeng Li, Mao Ye, Lihua Zhou, Nianxin Li, Siying Xiao, Song Tang, and Xiatian Zhu. Cloud object detector adaptation by integrating different source knowledge. *NeurIPS*, 37:25251–25283, 2024b.

Yu-Jhe Li, Xiaoliang Dai, Chih-Yao Ma, Yen-Cheng Liu, Kan Chen, Bichen Wu, Zijian He, Kris Kitani, and Peter Vajda. Cross-domain adaptive teacher for object detection. In *CVPR*, pp. 7581–7590, 2022c.

Yu-Jhe Li, Xiaoliang Dai, Chih-Yao Ma, Yen-Cheng Liu, Kan Chen, Bichen Wu, Zijian He, Kris Kitani, and Peter Vajda. Cross-domain adaptive teacher for object detection. In *CVPR*, pp. 7581–7590, 2022d.

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

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *ECCV*, pp. 38–55. Springer, 2024a.

Yabo Liu, Jinghua Wang, Chao Huang, Yaowei Wang, and Yong Xu. Cigar: Cross-modality graph reasoning for domain adaptive object detection. In *CVPR*, pp. 23776–23786, 2023.

Yajing Liu, Shijun Zhou, Xiyao Liu, Chunhui Hao, Baojie Fan, and Jiandong Tian. Unbiased faster r-cnn for single-source domain generalized object detection. In *CVPR*, pp. 28838–28847, 2024b.

Peggy Joy Lu, Chia-Yung Jui, and Jen-Hui Chuang. A privacy-preserving approach for multi-source domain adaptive object detection. In *ICIP*, pp. 1075–1079, 2023.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

Anton Mallasto and Aasa Feragen. Learning from uncertain curves: The 2-wasserstein metric for gaussian processes. *NeurIPS*, 30, 2017.

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

Xingang Pan, Xiaohang Zhan, Jianping Shi, Xiaoou Tang, and Ping Luo. Switchable whitening for deep representation learning. In *ICCV*, October 2019a.

Xingang Pan, Xiaohang Zhan, Jianping Shi, Xiaoou Tang, and Ping Luo. Switchable whitening for deep representation learning. In *ICCV*, pp. 1863–1871, 2019b.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pp. 8748–8763, 2021.

AN Rajagopalan et al. Improving robustness of semantic segmentation to motion-blur using class-centric augmentation. In *CVPR*, pp. 10470–10479, 2023.

Zhijie Rao, Jingcai Guo, Luyao Tang, Yue Huang, Xinghao Ding, and Song Guo. Srcd: Semantic reasoning with compound domains for single-domain generalized object detection. *IEEE Transactions on Neural Networks and Learning Systems*, 36(7):12497–12506, 2025.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *NeurIPS*, 28, 2015.

Kuniaki Saito, Yoshitaka Ushiku, Tatsuya Harada, and Kate Saenko. Strong-weak distribution alignment for adaptive object detection. In *CVPR*, pp. 6956–6965, 2019.

Zhenfeng Shao, Wenjing Wu, Zhongyuan Wang, Wan Du, and Chengyuan Li. Seaships: A large-scale precisely annotated dataset for ship detection. *IEEE Transactions on Multimedia*, 20(10):2593–2604, 2018.

Bo Sun, Banghuai Li, Shengcai Cai, Ye Yuan, and Chi Zhang. Fsce: Few-shot object detection via contrastive proposal encoding. In *CVPR*, pp. 7352–7362, 2021.

Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: A simple and strong anchor-free object detector. *IEEE TPAMI*, 44(4):1922–1933, 2020.

Vidit Vidit, Martin Engilberge, and Mathieu Salzmann. Clip the gap: A single domain generalization approach for object detection. In *CVPR*, pp. 3219–3229, 2023.

Vibashan VS, Poojan Oza, and Vishal M Patel. Instance relation graph guided source-free domain adaptive object detection. In *CVPR*, pp. 3520–3530, 2023.

Kunyu Wang, Xueyang Fu, Yukun Huang, Chengzhi Cao, Gege Shi, and Zheng-Jun Zha. Generalized uav object detection via frequency domain disentanglement. In *CVPR*, pp. 1064–1073, 2023.

Aming Wu and Cheng Deng. Single-domain generalized object detection in urban scene via cyclic-disentangled self-distillation. In *CVPR*, pp. 847–856, 2022.

Fan Wu, Jinling Gao, Lanqing Hong, Xinbing Wang, Chenghu Zhou, and Nanyang Ye. G-nas: Generalizable neural architecture search for single domain generalization object detection. In *AAAI*, volume 38, pp. 5958–5966, 2024.

Jiaxi Wu, Jiaxin Chen, Mengzhe He, Yiru Wang, Bo Li, Bingqi Ma, Weihao Gan, Wei Wu, Yali Wang, and Di Huang. Target-relevant knowledge preservation for multi-source domain adaptive object detection. In *CVPR*, pp. 5301–5310, 2022a.

Jiaxi Wu, Jiaxin Chen, Mengzhe He, Yiru Wang, Bo Li, Bingqi Ma, Weihao Gan, Wei Wu, Yali Wang, and Di Huang. Target-relevant knowledge preservation for multi-source domain adaptive object detection. In *CVPR*, pp. 5301–5310, 2022b.

Mingjun Xu, Lingyun Qin, Weijie Chen, Shiliang Pu, and Lei Zhang. Multi-view adversarial discriminator: Mine the non-causal factors for object detection in unseen domains. In *CVPR*, pp. 8103–8112, 2023.

Qinwei Xu, Ruipeng Zhang, Ya Zhang, Yanfeng Wang, and Qi Tian. A fourier-based framework for domain generalization. In *CVPR*, pp. 14383–14392, 2021.

Xiaoran Xu, Jiangang Yang, Wenyue Chong, Wenhui Shi, Shichu Sun, Jing Xing, and Jian Liu. Boosting single-domain generalized object detection via vision-language knowledge interaction. MM '25, pp. 131–140, 2025a. ISBN 9798400720352.

Xiaoran Xu, Jiangang Yang, Wenhui Shi, Siyuan Ding, Luqing Luo, and Jian Liu. Physaug: A physical-guided and frequency-based data augmentation for single-domain generalized object detection. In *AAAI*, volume 39, pp. 21815–21823, 2025b.

Xiaofei Yang, Hongwei She, Mengmeng Lou, Hui Ye, Jun Guan, Jianzhen Li, Zhengrong Xiang, Hao Shen, and Bin Zhang. A joint ship detection and waterway segmentation method for environment-aware of usvs in canal waterways. *IEEE Transactions on Automation Science and Engineering*, 22:2132–2144, 2024.

Xingxu Yao, Sicheng Zhao, Pengfei Xu, and Jufeng Yang. Multi-source domain adaptation for object detection. In *ICCV*, pp. 3273–3282, 2021.

Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *CVPR*, pp. 2636–2645, 2020.

Libo Zhang, Wenzhang Zhou, Heng Fan, Tiejian Luo, and Haibin Ling. Robust domain adaptive object detection with unified multi-granularity alignment. *IEEE TPAMI*, 46(12):9161–9178, 2024.

Xingguang Zhang and Chih-Hsien Chou. Source-free domain adaptation for video object detection under adverse image conditions. In *CVPR*, pp. 5010–5019, 2024.

Liang Zhao and Limin Wang. Task-specific inconsistency alignment for domain adaptive object detection. In *CVPR*, pp. 14217–14226, 2022.

Sicheng Zhao, Huizai Yao, Chuang Lin, Yue Gao, and Guiguang Ding. Multi-source-free domain adaptive object detection. *IJCV*, 132(12):5950–5982, 2024.

Yuyang Zhao, Zhun Zhong, Na Zhao, Nicu Sebe, and Gim Hee Lee. Style-hallucinated dual consistency learning for domain generalized semantic segmentation. In *ECCV*, pp. 535–552, 2022.

Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE TPAMI*, 45(4):4396–4415, 2022.

Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR*, 2021.

Zhengxia Zou, Keyan Chen, Zhenwei Shi, Yuhong Guo, and Jieping Ye. Object detection in 20 years: A survey. *Proceedings of the IEEE*, 111(3):257–276, 2023.

A Appendix

Due to the space limitation of the main paper, this supplementary materials provides additional information. The contents are organized as below:

- appendix B presents additional related work, covering DG for classification, DAOD, and a comparison of prior methods with ours in terms of DG paradigms and alignment schemes.
- appendix C provides further information details of our method, with a focus on the meta-optimization procedure.
- appendix D gives detailed descriptions of all datasets used in our experiments.
- appendix E reports additional comparisons with prior work, including MAML for DG method, further analyses of heterogeneous domain shifts, computational cost analysis, and a summary of DGOD evaluation coverage.
- appendix F includes extended analyses, including ablations on the auxiliary module, augmentation reliance, bounding boxes used in CDFA, hyperparameter sensitivity, and t-SNE visualizations of learned representations.
- appendix G reports class-wise performance of the compared methods on the benchmarks used in the main paper.
- appendix H presents the qualitative results.

B Additional related works

Domain generalization for classification. Domain generalization has a long history in image classification. Prior work is often grouped into three broad directions. (1) Domain augmentation approaches expand the variability of the training data by synthesizing or translating source images into additional styles or domains Xu et al. (2021); Huang et al. (2021). (2) Representation learning methods focus on learning features that remain stable across domains, thereby reducing sensitivity to domain-specific cues Li et al. (2018b). (3) Algorithmic training strategies, such as meta-learning-based formulations, train models in an episodic manner to mimic domain shift during optimization, encouraging the network to acquire transferable, domain-robust components Li et al. (2018a).

Domain adaptation object detection. Several methods in this regard attempt to bridge the distribution gap between the source and target domains by transferring knowledge from the source domain to unlabeled target domain data. Single-source free-domain adaptive detection methods make the assumption that there is only one source domain available for training the model to adapt to the target data Li et al. (2022b); VS et al. (2023); Zhang & Chou (2024); Hao et al. (2024). Meanwhile, multiple source domain approaches integrate different training source domains to adapt to the target domain Li et al. (2024b); Zhao et al. (2024); Lu et al. (2023). In different scenarios, when source domain data is unavailable but pretrained models on source domains are available, single Li et al. (2022b); VS et al. (2023); Zhang & Chou (2024); Hao et al. (2024) and multiple source-free Wu et al. (2022a); Belal et al. (2024); Yao et al. (2021); Wu et al. (2022b) domain adaptation methods are used to transfer knowledge from pretrained models on source domains to the target domain. We note that almost all of the methods mentioned above require access to unlabeled target domain data as well as an adaptation step to perform well in that domain. As a result, these techniques cannot be employed to address single-domain generalized object detection problems.

Multiple source domains generalization object detection. This setting assumes that there are multiple labeled source domains for training. To the best of our knowledge, only one method has been evaluated for generalized detection model across multiple source domains Lin et al. (2021) (*ICCV'21*), but does not provide the public code for reproducibility. Moreover, this method is also based on the simple class-level alignment, which is not directly designed for detection and may not be effective for the region-level domain shift in detection. Therefore, we skipped the comparison with this method in our experiments.

Comparison on DG schemes for object detection. table 9 compares methods along two axes that are often conflated in DGOD: (i) where the alignment signal is applied (feature-level vs. detection-head logits) and (ii) what entity is aligned (global class vs. instance/ROI vs. object region). This distinction matters because object detection is inherently *region-centric*: the same image can contain multiple objects of different classes, or even multiple instances of the same class with different appearances (sub-types, scales, poses, truncation/occlusion). Consequently, an image is more naturally represented by a *set of object regions* rather than a single global embedding.

Several classification-DG strategies adapted to detection (top block) primarily encourage domain-invariant representations at an image-level or class-level granularity, e.g., via normalization-based invariances, risk-regularization, or global class alignment (often implemented by matching class-wise statistics or distributions). While effective for classification, these approaches implicitly treat each sample as belonging to one (or a few) classes and therefore induce alignment over a small number of discrete class prototypes/points. In detection, however, such global alignment can under-specify the problem: the dominant shift often occurs at the object/ROI level (texture, scale, viewpoint, background context around the object), and this shift is not fully captured by aligning image-level or class-level aggregates.

Most dedicated DGOD methods that include alignment (middle block) rely on the detector's classification head outputs as the alignment carrier, typically using KL divergence (or variants) on per-class logits and/or contrastive objectives constructed from predicted categories. This yields either global class alignment (matching class distributions across domains) or instance-style alignment where an RoI is reduced to a class label and aligned accordingly. A few works move partially upstream by adding feature-level contrastive learning, but the supervision signal is still commonly derived from coarse partitions such as foreground vs. background or predicted class, which remains a point-wise / discrete view of what is being aligned.

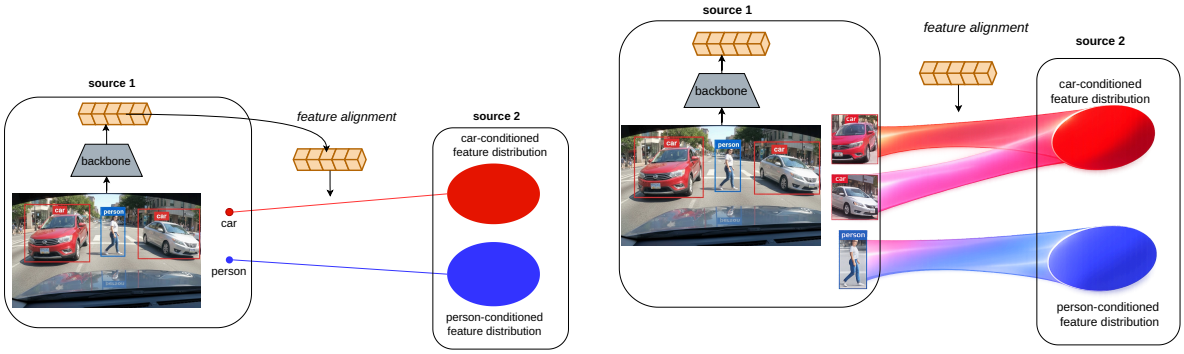
Table 9: **Positioning of alignment schemes in DG for object detection.** We summarize representative (i) classification-DG techniques adapted to detection, (ii) vision-only DGOD methods, and (iii) LLM/VLM-based DGOD methods, categorized by their DG scheme and the granularity and locus of alignment. Most prior DGOD alignment is driven by detection-head classification outputs (e.g., KL/contrastive on logits) and operates at global class or instance/foreground–background level, whereas our method performs object-conditional feature alignment on the *entire object region* using upstream features, making the alignment finer-grained and more directly tied to region-level domain shift.

Methods	**DG Scheme**	**Alignment scheme**	
		Alignment learning	**Instance**
Classification based DG methods for DGOD			
SW	Batch/Instance/Layer Normalization	-	-
IBN-Net	Feature normalization	-	-
IterNorm	Iterative Normalization	Iterative L2-squared norm	Class
ISW	Invariant Risk Minimization	L1 norm	Class
MASF	Domain-invariant representation learning	Meta learning + KL divergence + Contrastive learning (global class alignment)	Class
Vision-only dedicated DGOD methods			
SDGOD (CVPR'22)	Cyclic-Disentangled Self-Distillation	Detection head-level classification KL divergence + Feature-level contrastive learning (global/instance)	Class
SRCD (TNNLS'25)	Local–global semantic reasoning	Detection head-level classification KL divergence + Contrastive learning	Class
OADG (AAAI'24)	Domain-invariant learning	Detection head-level classification KL divergence + ROI-level contrastive learning (fore/background)	Class
G-NAS (AAAI'24)	Neural Architecture Search	No	No
UFR (CVPR'24)	Causal Attention Learning	Detection head-level classification KL divergence + Contrastive learning	Class
DivAlign (CVPR'24)	Aligning Diversified Domains	Detection head-level classification KL divergence + ROI IOU L2-squared norm	Class
Ours.	MAML object-conditional cross domain alignment	Feature-level entire object alignment + meta learning	**Object**
LLM/VLM based dedicated DGOD methods			
CLIP Gap (CVPR'24)	Vision-language model based semantic augmentation	Detection head-level classification contrastive loss	Class
PDOC (CVPR'24)	LLM based prompt-learning for visual feature	-	-
VLTDet (ACCV'24)	LLM based transfer learning	-	-
VLM-DGOD (MM'25)	LLM based cross-modal feature interaction	InfoNCE	Class
Diff. Guided (SD-2.1)(CVPR'25)	VLM based knowledge distillation	-	-
Diff. Guided (SD-1.5) (CVPR'25)	VLM based knowledge distillation	-	-

In contrast, our method performs alignment at the image feature level and conditions it on *object regions* (bounding-box-defined regions), not merely on class logits. Each image contributes a set of object features (one per object), and we align distributions over these object features across domains. This is fundamentally different from class-based alignment: instead of aligning a small number of class prototypes (discrete points), we align a richer object-conditioned feature set that can be multi-modal even within the same class (e.g., different viewpoints, scales, or sub-types). This makes the problem strictly more fine-grained and typically harder, but also more faithful to detection's region-level domain shift. Because entire image features lie upstream of the classification/regression heads, aligning them directly shapes the representations consumed by downstream detection losses. In contrast, aligning only the head outputs constrains the final predictions but may leave earlier ROI representations domain-biased, which can propagate to both classification and box regression errors. Our object-based feature alignment therefore targets the locus where cross-domain mismatch emerges for detection—the region-level representation—and complements the meta-learning procedure that explicitly trains under episodic domain shift.

Comparison on feature alignment. fig. 7 illustrates the different feature alignment approaches. In both paradigms, a backbone extracts semantic features from images and an alignment objective encourages domain-invariant representations across sources. The crucial difference is the conditioning metadata used to define what should be aligned.

In prior class-conditional alignment (left) (commonly used in classification task), the alignment meta is the image-level class identity (or class evidence aggregated from the detector head). As a result, each image is reduced to one (or a few) class-associated feature points. Alignment therefore becomes a point-wise (discrete) matching between class-labeled embeddings across domains. While this is natural for classification, it is mis-specified for object detection: treating an entire image as a class label discards the structure that matters for detection—where objects are and which pixels constitute each object instance—so the alignment



(a) Prior class-conditional feature alignment.

(b) Object conditional feature alignment (Ours).

Figure 7: Class-conditioned (discrete) vs. object-conditioned (set-valued) feature alignment. Prior alignment treats each image as a class-labeled point (or a few points), yielding point-wise matching that discards region structure. Our alignment uses all bounding boxes as metadata, representing each image as a set of regions features and aligning object-conditioned feature distributions across domains.

signal cannot faithfully address region-level shifts (e.g., object scale, pose, context, occlusion) that dominate detection errors.

In our object-conditional feature alignment (right), the alignment meta is the set of bounding boxes (object regions) present in the image. Instead of collapsing an image into a few class points, we represent each image by a set of region features extracted from all object boxes. Alignment is therefore set-valued / distributional: we match object-conditioned feature distributions across domains (e.g., car-conditioned, person-conditioned), where each class distribution is formed from multiple object instances within and across images. This preserves detection's essential information—instance identity, region content, and spatially grounded semantics—making the alignment directly applicable to DGOD. By aligning at the object-region granularity, our approach addresses the limitation of class-only alignment and better targets the true source of domain shift in detection: instance-level appearance and context variations rather than image-level category statistics.

C More details of our method

Detail of 2-Wasserstein distance. Here is the full formula for the 2-Wasserstein distance between two Gaussian distributions, which we use for feature alignment in our method:

$$W_2^2\big(\mathcal{N}(\boldsymbol{\mu}_a, \boldsymbol{\Sigma}_a), \mathcal{N}(\boldsymbol{\mu}_b, \boldsymbol{\Sigma}_b)\big) = \|\boldsymbol{\mu}_a - \boldsymbol{\mu}_b\|_2^2 + \mathrm{Tr}\Big(\boldsymbol{\Sigma}_a + \boldsymbol{\Sigma}_b - 2\big(\boldsymbol{\Sigma}_b^{1/2}\,\boldsymbol{\Sigma}_a\,\boldsymbol{\Sigma}_b^{1/2}\big)^{1/2}\Big) \tag{12}$$

where Tr denotes the matrix trace.

Meta optimization details. In addition to our method in the main paper, this supplementary section provides the exact meta-optimization used by our MAML-DGOD training procedure, including (i) how inner/outer losses are formed under the meta domain splitting in Algorithm 1, and (ii) how gradients are computed and accumulated using a first-order model-agnostic meta-learning (FOMAML) approximation.

We perform inner gradient computation as:

$$\Theta' \;\triangleq\; \Theta - \eta_{\text{inner}}\,\nabla_{\Theta}\mathcal{L}_{\text{inner}}(\Theta). \tag{13}$$

The meta-test (outer) loss is evaluated at Θ':

$$\mathcal{L}_{\text{outer}}(\Theta') \;\triangleq\; \mathcal{L}_{\text{det}}^{te}(\Theta') \;+\; \mathcal{L}_{\text{cda}}(\Theta') \;=\; \mathcal{L}_{\text{det}}^{te}(\Theta') \;+\; \lambda_{\text{cdfa}}\mathcal{L}_{\text{cdfa}}(\Theta') \;+\; \lambda_{W_2}\mathcal{L}_{\text{cda}}^{W_2}(\Theta'), \tag{14}$$

To reduce computational overhead, we adopt FOMAML, i.e., we do *not* backpropagate through the inner update (Eq. equation 13). Formally, the standard MAML meta-gradient would be:

$$\nabla_{\Theta}\mathcal{L}_{\text{outer}}(\Theta') = \left(\frac{\partial\Theta'}{\partial\Theta}\right)^{\top} \nabla_{\Theta'}\mathcal{L}_{\text{outer}}(\Theta'), \tag{15}$$

where $\frac{\partial\Theta'}{\partial\Theta}$ contains second-order terms. In FOMAML, we ignore $\frac{\partial\Theta'}{\partial\Theta}$ and use the approximation: $\nabla_{\Theta}\mathcal{L}_{\text{outer}}(\Theta') \approx \nabla_{\Theta'}\mathcal{L}_{\text{outer}}(\Theta')$. Practically, this corresponds to computing the inner-step update without creating a higher-order autograd graph (e.g., `create_graph=False`).

Analysis of the role of ψ and gradient accumulation. In Algorithm 1, the term ψ controls the trade-off between (i) the support detection signal and (ii) the meta-test episode objective. Importantly, this mixing is applied at the level of the meta-update direction. For a given meta-test domain D^{te}, we define:

$$g_{\text{inner}} \triangleq \nabla_{\Theta}\mathcal{L}_{\text{inner}}(\Theta), \qquad g_{\text{outer}} \triangleq \nabla_{\Theta'}\mathcal{L}_{\text{outer}}(\Theta') \quad \text{(FOMAML)}, \tag{16}$$

and combine them as:

$$g \triangleq \psi\, g_{\text{inner}} + (1-\psi)\, g_{\text{outer}}. \tag{17}$$

We accumulate g over all choices of $D^{te} \in \mathcal{D}_S$ into a gradient accumulator Υ_{Θ} and apply the outer update:

$$\Upsilon_{\Theta} \leftarrow \Upsilon_{\Theta} + g, \qquad \Theta \leftarrow \Theta - \eta_{\text{outer}}\, \Upsilon_{\Theta}. \tag{18}$$

D More details for the datasets

We aim to approximate a realistic deployment scenario in which an object detector is trained on different existing annotated datasets and is then expected to generalize to unseen datasets (or domains). To systematically assess this out-of-domain generalization ability, we construct evaluation protocols on several large-scale object detection benchmarks.

Diverse Weather. This benchmark is aim to evaluate DGOD in the presence of severe weather variations. Following Wu & Deng (2022), we adopt the Diverse Weather Dataset (DWD), which comprises five weather/domain conditions: Daytime Clear (DC), Daytime Foggy (DF), Night Sunny (NS), Night Rainy (NR), and Dusk Rainy (DR). The images are collected from BDD100K Yu et al. (2020), Foggy Cityscapes Cordts et al. (2016), and Adverse-Weather Hassaballah et al. (2021). In line with the standard Single-DGOD protocol, we use only DC as the source domain and evaluate on DF, NS, NR, and DR as unseen target domains. The DC split consists of 19,395 training images and 8,313 images for in-domain evaluation. The unseen domains contain 26,158 (NS), 2,494 (NR), 3,501 (DR), and 3,775 (DF) images, respectively, on which we report out-of-domain detection performance.

Cross Maritime. Maritime imagery introduces domain shifts that are not well captured by standard urban DGOD benchmarks, such as strong specular reflections on water surfaces, horizon-dominated geometry, wake clutter, small and distant vessels, and rapidly varying weather and sea states. These factors primarily perturb context and scale priors rather than only texture, making maritime scenarios a stringent test of robustness and background invariance, particularly for safety-critical unmanned surface vehicle (USV) applications. We therefore construct a Cross Maritime benchmark to measure generalization across sea, river, canal, and harbor environments with dynamic water and challenging visibility conditions. Concretely, we employ three public datasets captured in diverse settings: SeaShips Shao et al. (2018) (harbor in China), AboShips Iancu et al. (2021) (sea in Finland), and USV-canal Yang et al. (2024) (river/canal). SeaShips is split into 1,750 training, 1,750 validation, and 3,500 test images. AboShips is split into 3,010 training, 1,500 validation, and 2,369 test images. For USV-canal, we use 750 images for training and 321 for testing, while 391 images from SeePerSea Jeong et al. (2025) are used as the validation set. In each cross-domain setting, two of the three datasets are used as source domains, and the remaining dataset is treated as the unseen target. Models are trained on the training splits of the source domains, with their validation splits used for model selection, and final performance is reported on the test split of the unseen maritime domain.

Cross Locale. Road-damage detection across countries exhibits substantial domain shifts arising from differences in pavement materials, taxonomy definitions, camera setups, traffic layouts, and annotation protocols.

A Cross Locale benchmark makes these factors explicit, stressing transfer under changes in label semantics and visual appearance, and reflecting practical deployment scenarios where target-country annotations are scarce or unavailable. To this end, we use the RDD2022 benchmark Arya et al. (2024), which aggregates six country specific datasets: China (Cn), Czech Republic (Cz), Japan (Jp), India (In), Norway (No), and the United States (USA) with four shared damage categories: D00 (longitudinal crack), D10 (transverse crack), D20 (alligator crack), and D40 (other cracks, including potholes, block cracks, and repairs). For each country, we split the images into train/validation/test as follows: 1186/198/593 (Cn), 1697/283/849 (Cz), 4624/770/2312 (In), 6304/1050/3152 (Jp), 4897/816/2448 (No), and 2883/480/1442 (USA). In each Cross Locale experiment, we use 5 of the 6 countries as source domains and treat the remaining country as the unseen target domain. Models are trained on the training splits of the source domains, with their validation splits used for early stopping and model selection, and we report results on the test split of the held-out target country. The large number of domains and their diverse characteristics provide a rigorous test-bed for evaluating the robustness of domain generalization methods for object detection when leveraging different source domains.

Corruptions. Cityscapes-C (Corruptions) Michaelis et al. (2019) serves as a robustness benchmark that tests object detectors on systematically corrupted inputs. It includes 15 corruption operators, each provided at five severity settings, and organizes them into four groups: noise, blur, weather, and digital artifacts. The goal is to assess performance under out-of-distribution (OOD) perturbations. Notably, these corrupted variants are reserved strictly for testing and are not used during training. Following OADG Lee et al. (2024), we train the detector on the original Cityscapes training split (2,975 images) and use the validation split (500 images) for validation. For testing, Cityscapes-C provides corrupted versions of the validation images: for each of the 15 corruptions and each severity level, there are 500 images. Overall, we evaluate on 75 unseen corrupted test sets (15 corruptions $\times$ 5 severity levels).

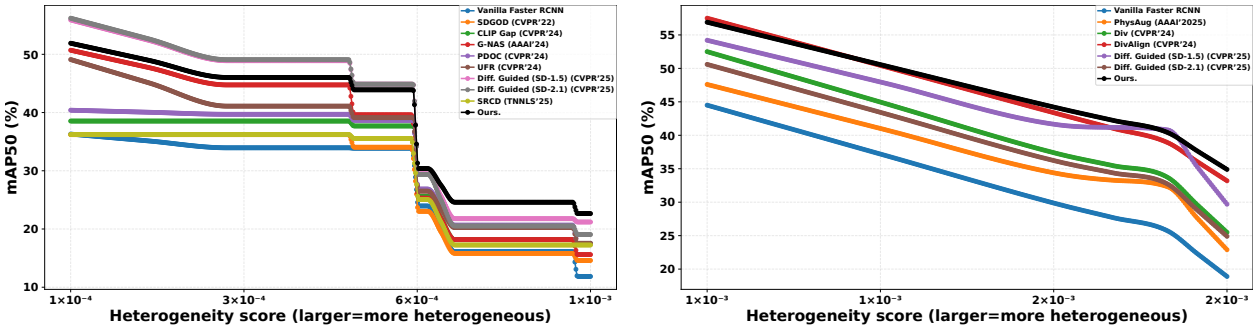
Real to Artistic. The benchmark includes four datasets. Pascal VOC serves as the source domain, containing real-world images across 20 object categories, while the remaining three datasets: Clipart, Watercolor, and Comic, provide stylized artwork and are treated as target domains. Clipart retains the full set of 20 VOC categories, whereas Watercolor and Comic each cover 6 categories that form subsets of the VOC label space. Following Danish et al. (2024), we merge the VOC2007 and VOC2012 trainval splits, obtaining 16,551 training images, and use the 4,992 images from the VOC2007 test split for in-domain evaluation. For the target domains, Clipart includes 1,000 images, and Watercolor and Comic each contain 2,000 images, with each dataset split evenly into train and test sets. As in He et al. (2025), we use all images from Clipart, Comic, and Watercolor as unseen domains for evaluation.

Finally, all datasets are preprocessed in VOC format. Attached zip files contain the annotation files for all datasets in VOC format, and the image files can be downloaded from the original dataset websites.

E More comparison results

Additional heterogeneous domain-shift robustness analysis. figs. 8a and 8b shows performance–heterogeneity curves for the compared detectors on Diverse Weather and Real to Artistic benchmarks. The curves reveal that our method preserves detection accuracy even on highly heterogeneous unseen domains, whereas existing DGOD approaches suffer pronounced performance drops, highlighting the superior robustness of our model to distribution shifts.

Computational cost analysis. We analyze the computational cost of our method in table 10, including training FLOPs,GPU memory and time on Diverse Weather benchmark (contains the largest training dataset among five benchmarks). Here, all reproducible methods are evaluated on the same setting and run on the same hardware (NVIDIA H200 GPU) over 2K iterations and using the same base detector (Faster R-CNN R101) for the fair comparison. We report the average training FLOPs,GPU memory and time, along with the percentage increase compared to the baseline Faster R-CNN. Compared with prior DGOD methods, our method shows comparable computational costs. The additional cost mainly comes from the MAML-based meta-learning strategy, while the proposed CDFA module only introduces marginal overhead. The auxiliary function g_ϕ adds a small amount of training FLOPs, GPU memory and training time, but it is negligible



(a) Diverse Weather. (b) Real to Artistic.

Figure 8: mAP50-heterogeneity curves on Diverse Weather and Real to Artistic.

Table 10: Computational cost analysis on Diverse Weathers benchmark. Added training time and GPU memory are measured relative to the baseline (Faster RCNN).

Method	**Train$_{\text{GPU}}$ Mem. (GB)**		**Train$_{\text{time}}$ (s)**		**Train$_{\text{GPU}}$ Mem. added (%)**		**Train$_{\text{time}}$ added (%)**		**Train FLOPs (G)**	
CLIP Gap Vidit et al. (2023) *(CVPR'23)*	$72_{\pm 0.8}$		$2776_{\pm 8}$		118.2		179.3		1587	
OADG Lee et al. (2024) *(AAAI'24)*	$65_{\pm 0.1}$		$1542_{\pm 5}$		96.7		55.1		881	
G-NAS Wu et al. (2024) *(AAAI'24)*	$63_{\pm 0.1}$		$2042_{\pm 10}$		90.1		104.4		1167	
DivAlign Danish et al. (2024) *(CVPR'24)*	$68_{\pm 0.1}$		$1792_{\pm 5}$		106.1		80.2		1024	
PhysAug Xu et al. (2025b) *(AAAI'25)*	$33_{\pm 0.2}$		$1697_{\pm 8}$		0		70.7		610	
Ours.	$65_{\pm 0.2}$		$1565_{\pm 10}$		96.7		57.4		895	
(detail)	MAML	g_ϕ	MAML	g_ϕ	MAML	g_ϕ	MAML	g_ϕ	MAML	g_ϕ
	$63.7_{\pm 0.1}$	$1.3_{\pm 0.1}$	$1557_{\pm 6}$	$8_{\pm 1}$	93.0	3.9	56.6	0.8	890.4	4.6

compared to the overall meta-learning cost. Overall, our method achieves a favorable trade-off between performance and computational cost.

Table 11: **DGOD evaluation coverage and reporting of computational analysis.** We summarize representative DGOD methods by the number of benchmarks and datasets evaluated, the detector architectures used as testbeds, and whether the paper provides an explicit computational cost analysis (e.g., training cost (time and memory)). Here, ×indicates not reported, and ✓indicates reported.

DGOD Method	**Num. of Benchmarks**	**Num. Of Datasets**	**Tested existing detectors**	**Computational cost analysis**
SDGOD (CVPR'22)	1	5	Faster-RCNN	×
SRCD (TNNLS'25)	4	9	Faster-RCNN	×
CLIP Gap (CVPR'23)	1	5	Faster-RCNN	×
VLTDet (ACCV'24)	1	5	Faster-RCNN	×
PDOC (CVPR'24)	1	5	Faster-RCNN	×
G-NAS (AAAI'24)	1	5	Faster-RCNN	×
OA-Mix (AAAI'24)	2	7	Faster-RCNN	×
OA-DG (AAAI'24)	2	7	Faster-RCNN, YOLOv3	×
UFR (CVPR'24)	1	5	Faster-RCNN	×
Div (CVPR'24)	2	9	Faster-RCNN, FCOS	×
DivAlign (CVPR'24)	2	9	Faster-RCNN	×
PhysAug (AAAI'25)	2	7	Faster-RCNN	×
Diff. Guided (SD-1.5) (CVPR'25)	6	15	Faster-RCNN	×
Diff. Guided (SD-2.1) (CVPR'25)	6	15	Faster-RCNN	×
VLM-DGOD (MM'25)	2	7	Faster-RCNN	×
Ours.	5	20	Faster-RCNN, YOLOX, DETR, Grounding DINO	✓

Comparison on DGOD evaluation coverage. table 11 highlights some practical gaps in current DGOD evaluation. First, most methods are assessed on limited scopes (often a single benchmark with five datasets) and predominantly on Faster R-CNN, leaving open whether gains generalize across detector families (one-stage, transformer-based, or vision-language detectors). Second, computational evidence is frequently missing: despite additional components such as augmentation pipelines, NAS/search, or diffusion/VLM modules, many works do not quantify the resulting overhead (memory, or training-time increase), which complicates

fair cost–accuracy comparisons and real-world adoption. In contrast, our evaluation spans a broader set (5 benchmarks / 20 datasets) and multiple detector paradigms (Faster R-CNN, YOLOX, DETR, Grounding DINO), and we explicitly report computational analysis, enabling a more transparent assessment of generalization benefits versus efficiency trade-offs.

F Additional analysis of our method

Analysis of impact of auxiliary function. We utilize different networks to analyze the impact of auxiliary functions on the performance of our model. Here we utilize pretrained ResNet18, VGG-19, MobileNetV2, and Inception-V3 on ImageNet for training auxiliary function g_ϕ on ground truth bboxes of the source domain. Table 12 shows the results of our model on the Cross Locale benchmarks. In general, auxiliary networks with ResNet18 and Inception-v3 lead to higher performance than other networks. We observe that the performance of our model is relative to the accuracy of auxiliary function g_ϕ.

Table 12: Ablation study: impact of auxiliary function g_ϕ on performance of our method using Cross Locale benchmark. *Avg. F1-score* is the average F1-score performance of g_ϕ on evaluating ground truth bboxes of meta-test domain. mAP50 is the performance of our detection on the unseen domain using g_ϕ.

g_ϕ	Cz&In&Jp&No&USA to Cn		Cn&In&Jp&No&USA to Cz		Cn&Cz&Jp&No&USA to In		Cn&Cz&In&No&USA to Jp		Cn&Cz&In&Jp&USA to No		Cn&Cz&In&Jp&No to USA	
	Avg. F1-score	mAP50	Avg. F1-score	mAP50	Avg. F1-score	mAP50	Avg. F1-score	mAP50	Avg. F1-score	mAP50	Avg. F1-score	mAP50
ResNet18	***0.43***	**34.9**	***0.38***	**20.5**	*0.27*	17.8	***0.38***	**29.8**	***0.35***	**17.9**	*0.36*	41.6
VGG-19	*0.41*	34.1	*0.37*	19.7	***0.3***	18	*0.37*	29.1	*0.33*	17.3	*0.34*	39.6
MobileV3	***0.43***	34.8	*0.35*	20.3	*0.28*	17.1	*0.37*	28.5	*0.34*	17.9	*0.36*	40.7
Inception-v3	*0.41*	34.5	***0.38***	20.4	***0.3***	**18.2**	*0.37*	29.4	***0.35***	**17.9**	***0.37***	**41.8**

Analysis of augmentation reliance. Prior DGOD methods apply augmentation to the training data while, whereas we use it during meta-testing to *simulate* ***large*** *domain shift*. Here, we analyze the effect of applying augmentations $\mathcal{A}$ to the meta-train domain during meta-optimization. table 13 reports the performance of both the auxiliary function on meta-test domain and the full detector on the unseen target domain. The results indicate that when $\mathcal{A}$ is applied to the meta-train domain D^{tr}, the auxiliary function which trained solely on ground-truth bboxes, experiences a substantial degradation in performance, which in turn leads to a noticeable drop in detection accuracy. Therefore, in our final MAML-DGOD configuration, we refrain from applying augmentations $\mathcal{A}$ to the meta-train domain.

Table 13: Effect of applying augmentation techniques $\mathcal{A}$ to the meta-train domain in our method, evaluated on the Cross Locale benchmark. F1-score reports the performance of g_ϕ on ground-truth bboxes from the meta-test domain, while mAP50 measures the detection performance of our model on the same unseen domain. $\times$: $\mathcal{A}$ applied to the meta-train domain D^{tr}; $\checkmark$: no augmentation applied to D^{tr}.

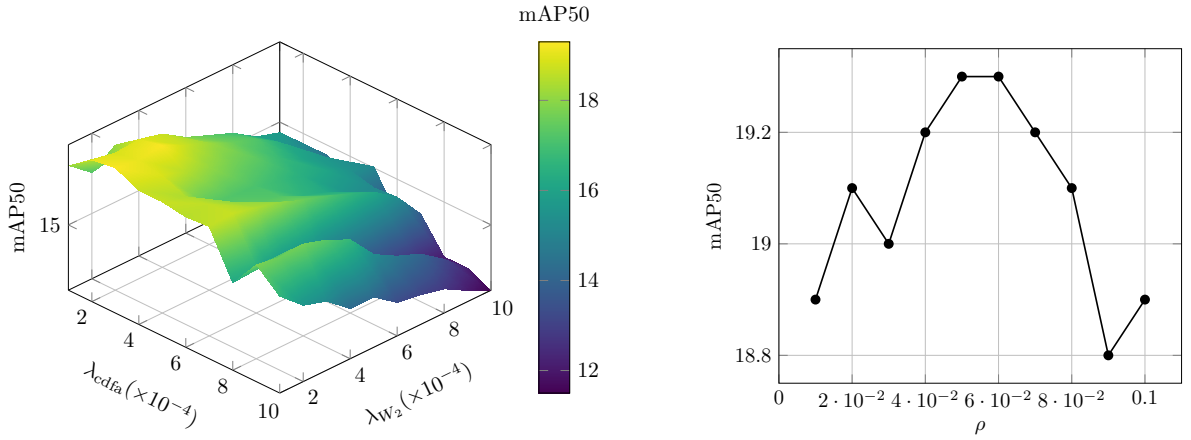
Experiment	w/o ${}^{\mathcal{A}}D^{tr}$	Avg. F1-score	mAP50
Cn&Cz&In&Jp&USA to No	$\times$	0.32	16.7
	$\checkmark$	0.35	17.9
Cn&In&Jp&No&USA to Cz	$\times$	0.36	18.9
	$\checkmark$	0.38	20.5

Used bounding boxes in CDFA: We analyze the effect of using predicted bboxes in CDFA by replacing GT bboxes with predicted bboxes for training CDFA. Here, we use the predicted bboxes from the base detector trained on source domains. As shown in table 14, using predicted bboxes significantly degrades performance.

Table 14: Effectiveness of sensitivity of used bboxes in CDFA to the performance.

Used bboxes in CDFA	**SA&SS2SC**	**SA&SC2SS**	**SC&SS2SA**
GT bboxes	**55.7**	**59**	**19.3**
Predicted bboxes	52.8	54.7	16.1

Analysis of loss weights. fig. 9a presents the sensitivity analysis of the loss weights λ_{cdfa} and λ_{W_2}. All experiments are conducted on the Cross Maritime benchmark which experiment is SS&SC to SA. Our method attains its highest mAP50 when λ_{cdfa} and λ_{W_2} are set to 3×10^{-4} and 1×10^{-4}, respectively. When either weight is further increased (beyond approximately 6×10^{-4}), the performance degrades notably. Considering both peak mAP50 and overall stability, we therefore adopt $\lambda_{\text{cdfa}} = 3 \times 10^{-4}$ and $\lambda_{W_2} = 1 \times 10^{-4}$ as our final configuration.



(a) Sensitivity to λ_{cdfa} and λ_{W_2} (SS&SC→SA).

(b) Sensitivity to shrinkage ρ (SS&SC→SA).

Figure 9: Additional hyperparameter ablations on Cross Maritime (SS&SC→SA).

Shrinkage factor analysis. We further analyze the influence of the shrinkage factor on the performance of our method by varying the hyperparameter ρ in the range $[0.01, 0.1]$. As shown in fig. 9b, the model attains its best performance for ρ values around 0.05–0.06. Based on this observation, we fix ρ to a single value within this range in our final experimental setup.

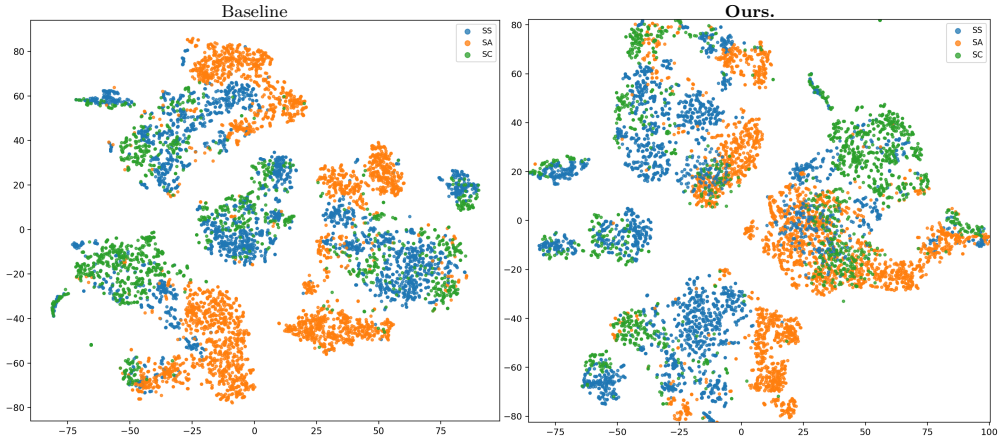
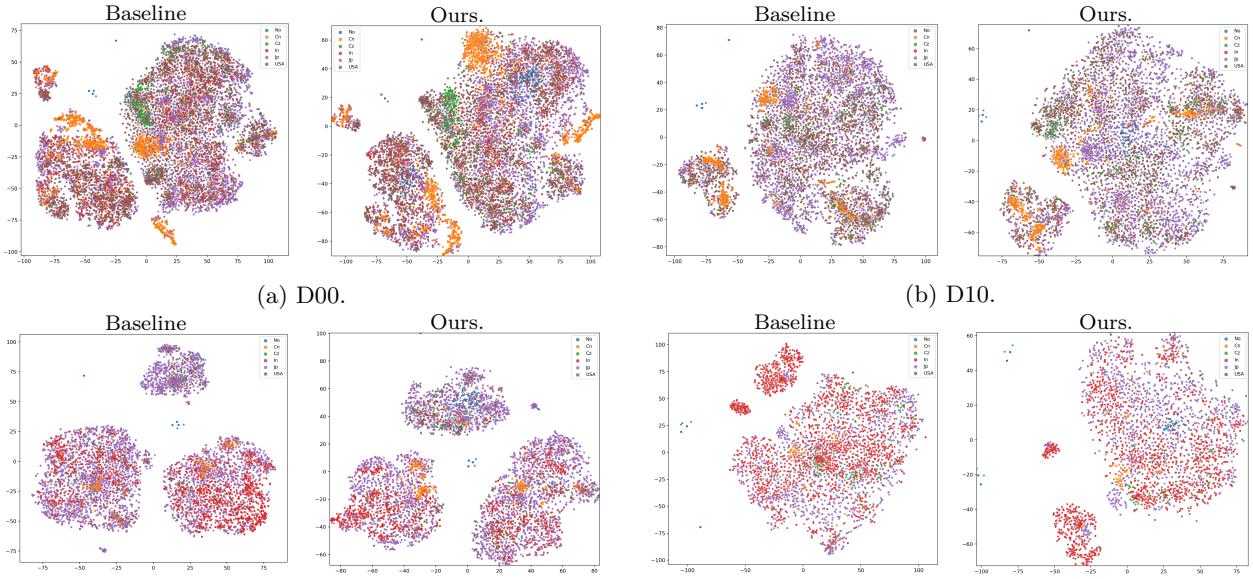


Figure 10: t-SNE visualization of features learned by baseline (*i.e.*, Faster-RCNN) and our method in Cross Maritime (SC&SS to SA).

Analysis of learning embeddings. We illustrate the learned features using t-SNE Maaten & Hinton (2008) on the Cross Maritime (experiment SC&SS to SA) and Cross Locale benchmarks (experiment Cn&Cz&In&Jp&USA to No). fig. 10 shows the feature distribution of ship; and fig. 11 shows the feature distribution of 4 object classes which are D00, D10, D20, and D40 across source and unseen domains. The learned features of baseline distributes separate in comparison to our method. Furthermore, our method effectively maps features into diverse spaces and aligns various source domains to enhance the model's generalization.



(a) D00. (b) D10.

(c) D20. (d) D40.

Figure 11: t-SNE visualization of features learned by baseline (*i.e.*, Vanilla Faster-RCNN Ren et al. (2015)) and our method in Cross Locale (Cn&Cz&In&Jp&USA to No).

G Class-wise performance

Similar to the compared methods, we report the detailed average precision (AP) class-wise performance on the Diverse Weather, Cross Locale and Real to Artistic benchmarks. In all tables of this section, bold and underline denote the best and second-best results; '-' indicates the lack of a performance report; and ▷ indicates the results are obtained directly from referred works. Note that we report only the results of DGOD methods here. Detailed class-wise results for classification-based methods on the Diverse Weather benchmark are available in Wu & Deng (2022), while those for domain adaptation methods on the Real-to-Artistic benchmark can be found in He et al. (2025). Finally, we omit the class-wise performance on the Corruptions benchmark, since no prior work has reported such results for this benchmark. For completeness, however, we include them in the attached code.

tables 15 to 18 present the average precision (AP) of object classes on the Diverse-Weather benchmark; table 19 summarize the AP of four object classes on the Cross Locale benchmark and tables 21 to 23 present the AP of object classes on the Real to Artistic benchmark. *Diff. Guided (SD-1.5 and SD-2.1)* He et al. (2025) achieves the best performance on Daytime Foggy and Night Sunny. In contrast, for the more challenging unseen domains Dusk Rainy and Night Rainy, which exhibit a higher level of heterogeneity with respect to the source domain, our method outperforms competing approaches on the majority of object classes, including bike, car, person, and rider. On the Cross Locale benchmark, our method is comparable to prior methods on Cz and Cn, while yielding consistently better performance on In, Jp, No, and USA.

We additionally report the performance of the models on the validation sets of the source domains in the Cross Maritime benchmark, as summarized in table 20. The results demonstrate that our domain-alignment strategy consistently improves the performance on each validation dataset and yields a more robust model than other methods.

Table 15: Class-wise performance of detection models on Night Rainy of Diverse Weather benchmark.

Methods	**bus**	**bike**	**car**	**motor**	**person**	**rider**	**truck**	**mAP50**
Vanilla Faster RCNN Ren et al. (2015)	21.3	7.7	28.8	6.1	8.9	10.3	16	14.2
▷SDGOD Wu & Deng (2022) *(CVPR'22)*	24.4	11.6	29.5	0.4	9.9	10.6	17.6	16.6
▷SRCD Rao et al. (2025)*(TNNLS'25)*	26.5	12.9	32.4	0.8	10.2	12.5	24	17
▷CLIP Gap Vidit et al. (2023) *(CVPR'24)*	28.6	12.1	26.1	9.2	12.3	9.6	22.9	18.7
▷PDOC Li et al. (2024a) *(CVPR'24)*	25.6	12.1	35.8	**10.1**	14.2	12.9	22.9	19.2
▷OA-Mix Lee et al. (2024) *(AAAI'24)*	-	-	-	-	-	-	-	14.8
▷OADG Lee et al. (2024) *(AAAI'24)*	-	-	-	-	-	-	-	16.8
▷G-NAS Wu et al. (2024) *(AAAI'24)*	28.6	9.8	38.4	0.1	13.8	9.8	21.4	17.4
▷UFR Liu et al. (2024b) *(CVPR'24)*	29.9	11.8	36.1	9.4	13.1	10.5	23.3	19.2
▷Div Danish et al. (2024) *(CVPR'24)*	-	-	-	-	-	-	-	22
▷DivAlign Danish et al. (2024) *(CVPR'24)*	-	-	-	-	-	-	-	24.1
▷PhysAug Xu et al. (2025b) *(AAAI'25)*	-	-	-	-	-	-	-	23.1
▷Diff. Guided (SD-1.5) He et al. (2025) *(CVPR'25)*	35.4	12.7	46.2	3.2	13.8	10.7	29.7	21.7
▷Diff. Guided (SD-2.1) He et al. (2025) *(CVPR'25)*	34.4	7.8	43.3	2.2	14.3	7.5	30.3	20
▷VLM-DGOD Xu et al. (2025a) *(MM'25)*	-	-	-	-	-	-	-	23.8
Ours.	34.4	**13.8**	**51**	1.4	**21.3**	**15.6**	**31.5**	**25.1**

Table 16: Class-wise performance of detection models on Dusk Rainy of Diverse Weather benchmark.

Methods	**bus**	**bike**	**car**	**motor**	**person**	**rider**	**truck**	**mAP50**
Vanilla Faster RCNN Ren et al. (2015)	34.2	21.8	47.9	16	22.9	18.5	34.9	28
▷SDGOD Wu & Deng (2022) *(CVPR'22)*	37.1	19.6	50.9	13.4	19.7	16.3	40.7	28.2
▷SRCD Rao et al. (2025)*(TNNLS'25)*	39.5	21.4	50.6	11.9	20.1	17.6	40.5	28.8
▷CLIP Gap Vidit et al. (2023) *(CVPR'24)*	37.8	22.8	60.7	16.8	26.8	18.7	42.4	32.3
▷PDOC Li et al. (2024a) *(CVPR'24)*	39.4	25.2	60.9	**20.4**	29.9	16.5	43.9	33.7
▷OA-Mix Lee et al. (2024) *(AAAI'24)*	-	-	-	-	-	-	-	33.8
▷OADG Lee et al. (2024) *(AAAI'24)*	-	-	-	-	-	-	-	33.9
▷G-NAS Wu et al. (2024) *(AAAI'24)*	**44.6**	22.3	66.4	14.7	32.1	19.6	45.8	35.1
▷UFR Liu et al. (2024b) *(CVPR'24)*	37.1	21.8	67.9	16.4	27.4	17.9	43.9	33.2
▷Div Danish et al. (2024) *(CVPR'24)*	-	-	-	-	-	-	-	37
▷DivAlign Danish et al. (2024) *(CVPR'24)*	-	-	-	-	-	-	-	38.1
▷PhysAug Xu et al. (2025b) *(AAAI'25)*	-	-	-	-	-	-	-	41.2
▷Diff. Guided (SD-1.5) He et al. (2025) *(CVPR'25)*	43.1	23.9	73.9	13.4	33.2	21.1	52.3	37.4
▷Diff. Guided (SD-2.1) He et al. (2025) *(CVPR'25)*	**44.6**	22.5	73.1	15.7	31.7	19.3	**52.6**	37.1
▷VLM-DGOD Xu et al. (2025a) *(MM'25)*	-	-	-	-	-	-	-	41.3
Ours.	44.4	**29**	**74.3**	20.1	**42.4**	**24.5**	49.3	**41.5**

Table 17: Class-wise performance of detection models on Night Sunny of Diverse Weather benchmark.

Methods	**bus**	**bike**	**car**	**motor**	**person**	**rider**	**truck**	**mAP50**
Vanilla Faster RCNN Ren et al. (2015)	43.5	31.2	49.8	17.5	36.3	29.2	43.1	35.8
▷SDGOD Wu & Deng (2022) *(CVPR'22)*	40.6	35.1	50.7	19.7	34.7	32.1	43.4	36.6
▷SRCD Rao et al. (2025)*(TNNLS'25)*	43.1	32.5	52.3	20.1	34.8	31.5	42.9	36.7
▷CLIP Gap Vidit et al. (2023) *(CVPR'24)*	37.7	34.3	58	19.2	37.6	28.5	42.9	36.9
▷PDOC Li et al. (2024a) *(CVPR'24)*	40.9	35	59	21.3	40.4	29.9	42.9	38.5
▷OA-Mix Lee et al. (2024) *(AAAI'24)*	-	-	-	-	-	-	-	38.6
▷OADG Lee et al. (2024) *(AAAI'24)*	-	-	-	-	-	-	-	38
▷G-NAS Wu et al. (2024) *(AAAI'24)*	46.9	40.5	67.5	26.5	50.7	35.4	47.8	45
▷UFR Liu et al. (2024b) *(CVPR'24)*	43.6	38.1	66.1	14.7	49.1	26.4	47.5	40.8
▷Div Danish et al. (2024) *(CVPR'24)*	-	-	-	-	-	-	-	39.4
▷DivAlign Danish et al. (2024) *(CVPR'24)*	-	-	-	-	-	-	-	42.5
▷PhysAug Xu et al. (2025b) *(AAAI'25)*	-	-	-	-	-	-	-	44.9
▷Diff. Guided (SD-1.5) He et al. (2025) *(CVPR'25)*	51	42.8	72.2	27.5	55.9	39.5	52	48.7
▷Diff. Guided (SD-2.1) He et al. (2025) *(CVPR'25)*	**51.3**	**43.6**	**72.3**	**27.6**	**56.2**	**40.2**	**53.7**	**49.3**
▷VLM-DGOD Xu et al. (2025a) *(MM'25)*	-	-	-	-	-	-	-	45
Ours.	45.7	39.3	70.5	22.1	51.9	35.7	47.5	45

Table 18: Class-wise performance of detection models on Daytime Foggy of Diverse Weather benchmark.

Methods	**bus**	**bike**	**car**	**motor**	**person**	**rider**	**truck**	**mAP50**
Vanilla Faster RCNN Ren et al. (2015)	34.5	29.6	49.3	26.2	33	35.1	26.7	33.5
▷SDGOD Wu & Deng (2022) *(CVPR'22)*	32.9	28	48.8	29.8	32.5	38.2	24.1	33.5
▷SRCD Rao et al. (2025)*(TNNLS'25)*	36.4	30.1	52.4	31.3	33.4	40.1	27.7	35.9
▷CLIP Gap Vidit et al. (2023) *(CVPR'24)*	36.1	34.3	58	33.1	39	43.9	25.1	38.5
▷PDOC Li et al. (2024a) *(CVPR'24)*	36.1	34.5	58.4	33.3	40.5	44.2	26.2	39.1
▷OA-Mix Lee et al. (2024) *(AAAI'24)*	-	-	-	-	-	-	-	38.1
▷OADG Lee et al. (2024) *(AAAI'24)*	-	-	-	-	-	-	-	38.3
▷G-NAS Wu et al. (2024) *(AAAI'24)*	32.4	31.2	57.7	31.9	38.6	38.5	24.5	36.4
▷UFR Liu et al. (2024b) *(CVPR'24)*	36.9	35.8	61.7	33.7	39.5	42.2	27.5	39.6
▷Div Danish et al. (2024) *(CVPR'24)*	-	-	-	-	-	-	-	35.6
▷DivAlign Danish et al. (2024) *(CVPR'24)*	-	-	-	-	-	-	-	37.2
▷PhysAug Xu et al. (2025b) *(AAAI'25)*	-	-	-	-	-	-	-	40.8
▷Diff. Guided (SD-1.5) He et al. (2025) *(CVPR'25)*	**39.3**	35.8	**69.4**	**37.7**	**48.8**	**49.7**	32.3	**44.7**
▷Diff. Guided (SD-2.1) He et al. (2025) *(CVPR'25)*	38.8	**36.4**	68.9	37.4	48.6	49.6	**33.4**	**44.7**
▷VLM-DGOD Xu et al. (2025a) *(MM'25)*	-	-	-	-	-	-	-	42.1
Ours.	34.1	34.3	65.3	35.3	45.8	44.2	29.1	41.9

Table 19: Class-wise performance of detection models on Cross Locale benchmark under leave-one-domain-out evaluation.

Methods	**Cn&Cz&In&Jp&No → USA**					**Cn&Cz&In&Jp&USA → No**				
	D00	**D10**	**D20**	**D40**	**mAP50**	**D00**	**D10**	**D20**	**D40**	**mAP50**
Vanilla Faster RCNN Ren et al. (2015)	48.3	29.8	39.1	3.5	30.175	10.6	9.1	9.8	1.5	7.75
CLIP Gap Vidit et al. (2023) *(CVPR'24)*	51.6	33.4	42.3	6.4	33.425	11.4	10.4	10.4	2.5	8.675
OA-Mix Lee et al. (2024) *(AAAI'24)*	53.7	35.5	42.7	9.8	35.425	11.5	9.7	10.3	4.2	8.925
G-NAS Wu et al. (2024) *(AAAI'24)*	53.3	38.7	43.2	10.6	36.45	11.7	10.6	11.1	3.9	9.325
DivAlign Danish et al. (2024) *(CVPR'24)*	54.3	39.1	43	11.8	37.05	14.5	12.7	11.7	4.6	10.875
PhysAug Xu et al. (2025b) *(AAAI'25)*	56.8	41.7	43.1	12.2	38.45	11.9	11.2	10.8	9.1	10.75
Ours.	**58.8**	**42.7**	**47.1**	**18**	**41.65**	**25.9**	**16.9**	**17.7**	**11.1**	**17.9**

Methods	**Cn&Cz&In&No&USA → Jp**					**Cn&Cz&Jp&No&USA → In**				
	D00	**D10**	**D20**	**D40**	**mAP50**	**D00**	**D10**	**D20**	**D40**	**mAP50**
Vanilla Faster RCNN Ren et al. (2015)	15.4	12	36.3	3.9	16.9	7.6	1.2	13.3	10	8.025
CLIP Gap Vidit et al. (2023) *(CVPR'24)*	19.2	15.3	37.2	8.3	20	8.2	1	16.1	10.9	9.05
OA-Mix Lee et al. (2024) *(AAAI'24)*	22.4	16.2	38.2	11.5	22.075	9.3	1.3	18.7	11.9	10.3
G-NAS Wu et al. (2024) *(AAAI'24)*	24.7	16.4	37.8	11.6	22.625	9.4	1.4	19.2	12.1	10.525
DivAlign Danish et al. (2024) *(CVPR'24)*	26.1	19.4	40.1	15.2	25.2	13.1	2.3	22.8	12.8	12.75
PhysAug Xu et al. (2025b) *(AAAI'25)*	28.8	22.3	40.2	23.8	28.775	14.4	1.7	25.5	13.7	13.825
Ours.	**28**	**23.2**	**41.3**	**26.6**	**29.775**	**14.7**	**11.5**	**29.1**	**16.1**	**17.85**

Methods	**Cn&In&Jp&No&USA → Cz**					**Cz&In&Jp&No&USA → Cn**				
	D00	**D10**	**D20**	**D40**	**mAP50**	**D00**	**D10**	**D20**	**D40**	**mAP50**
Vanilla Faster RCNN Ren et al. (2015)	16	23.4	4.8	1.7	11.475	22.4	32.8	45.3	4.8	26.325
CLIP Gap Vidit et al. (2023) *(CVPR'24)*	17.1	26.1	6.7	3.1	13.25	22.5	35.6	46.1	5.6	27.45
OA-Mix Lee et al. (2024) *(AAAI'24)*	17.7	27.5	9.4	4.3	14.725	23.7	36.7	47.6	8.5	29.125
G-NAS Wu et al. (2024) *(AAAI'24)*	18.2	28.1	9.8	4.5	15.15	23.5	37.4	48.1	10.1	29.775
DivAlign Danish et al. (2024) *(CVPR'24)*	19.6	29	10.5	4.8	15.975	25.2	39.2	48.2	8.9	30.375
PhysAug Xu et al. (2025b) *(AAAI'25)*	20.8	**30.6**	14.3	5.1	17.7	22.9	**41.5**	**50.6**	12	31.75
Ours.	**21.2**	28.9	**18.8**	**13.2**	**20.525**	**27.9**	39.7	49.2	**22.9**	**34.925**

Table 20: Performance of detection methods on validation sets of source domains and on unseen domains using Cross Maritime benchmark.

Methods	**Data setting**									**Avg.**
	SA&SS to SC			**SA&SC to SS**			**SC&SS to SA**			
	SA	**SS**	**SC**	**SA**	**SC**	**SS**	**SC**	**SS**	**SA**	
Vanilla Faster RCNN Ren et al. (2015)	54.4	89.6	38.4	54.3	75.2	36.5	68.5	89	10.5	37.8
CLIP Gap Vidit et al. (2023) *(CVPR'24)*	58.18	90.33	45.93	58.5	79.57	36.11	79.16	89.39	11.61	42.7
OA-Mix Lee et al. (2024) *(AAAI'24)*	61.9	88.2	46.82	60.3	80.23	41.28	78.4	88.52	11.7	45
G-NAS Wu et al. (2024) *(AAAI'24)*	62.4	89.3	47.4	64.2	80.7	43.12	80.23	89.1	11.9	46
DivAlign Danish et al. (2024) *(CVPR'24)*	63.4	89.1	49.6	63.5	81.3	54.8	80.1	88.9	12.3	51.3
PhysAug Xu et al. (2025b) *(AAAI'25)*	66.3	88.8	54.4	68	84.2	50.2	82.5	89.7	14.7	53
Ours	68.3	89.9	**55.7**	68.2	87.7	**59**	88	89.8	**19.3**	**56.8**

Table 21: Class-wise performance of detection methods on Clipart of Real to Artistic benchmark. ◇: results are obtained directly from He et al. (2025).

Methods	**aero.**	**bike**	**bird**	**boat**	**bottle**	**bus**	**car**	**cat**	**chair**	**cow**	**table**	**dog**	**horse**	**bike**	**psn.**	**plant.**	**sheep**	**sofar**	**train**	**tv**	**mAP50**
◇Vanilla Faster RCNN Ren et al. (2015)	19.9	51.6	17	21.9	27.2	49.6	25.5	9.1	35.1	9.1	25.4	3	29.2	48.9	30.1	40.3	9.1	6.7	35.2	21	25.7
◇Div Danish et al. (2024)*(CVPR'24)*	29.3	50.3	23.4	35.3	45.3	49.8	33.4	10.6	43.3	22.3	31.6	4.5	32.9	51.9	40.2	51.1	18.2	29.6	42.3	28.5	33.7
◇DivAlign Danish et al. (2024)*(CVPR'24)*	34.4	**64.4**	22.7	27	45.6	59.2	32.9	7	46.8	**55.8**	28.9	14.5	**44.4**	58	55.2	52.1	14.8	**38.4**	42.5	33.9	38.9
PhysAug Xu et al. (2025b)*(AAAI'25)*	32.1	46.8	23.2	25.6	39.9	58.8	33.1	18.2	43	18.2	27.6	15	21.7	43.3	32.1	48	9.1	29.3	**44.5**	35.6	32.3
◇Diff. Guided (SD-2.1) He et al. (2025)*(CVPR'25)*	19.3	57.8	28.4	**37.4**	**57.8**	**81.3**	**46.3**	3.8	**57.8**	27.2	28.3	19.6	42.5	50.9	57.8	**59.8**	15.6	36	37.7	**50.5**	**40.8**
◇Diff. Guided (SD-1.5) He et al. (2025)*(CVPR'25)*	25.6	40.2	26.2	25.7	44.8	72.9	34.8	3.8	46.3	14	26.6	7.5	27.2	57.1	48.4	56.4	6.8	25.3	24.5	39.2	32.7
Ours.	**46.5**	51.1	**32.1**	26.6	43.3	42.2	34.6	**19.3**	46.2	37.6	**35.7**	**26.9**	35.3	**72.6**	**70.5**	43.3	**33.3**	22.7	43.1	44.8	40.4

Table 22: Class-wise performance of detection methods on Comic of Real to Artistic benchmark. ◇: results are obtained directly from He et al. (2025).

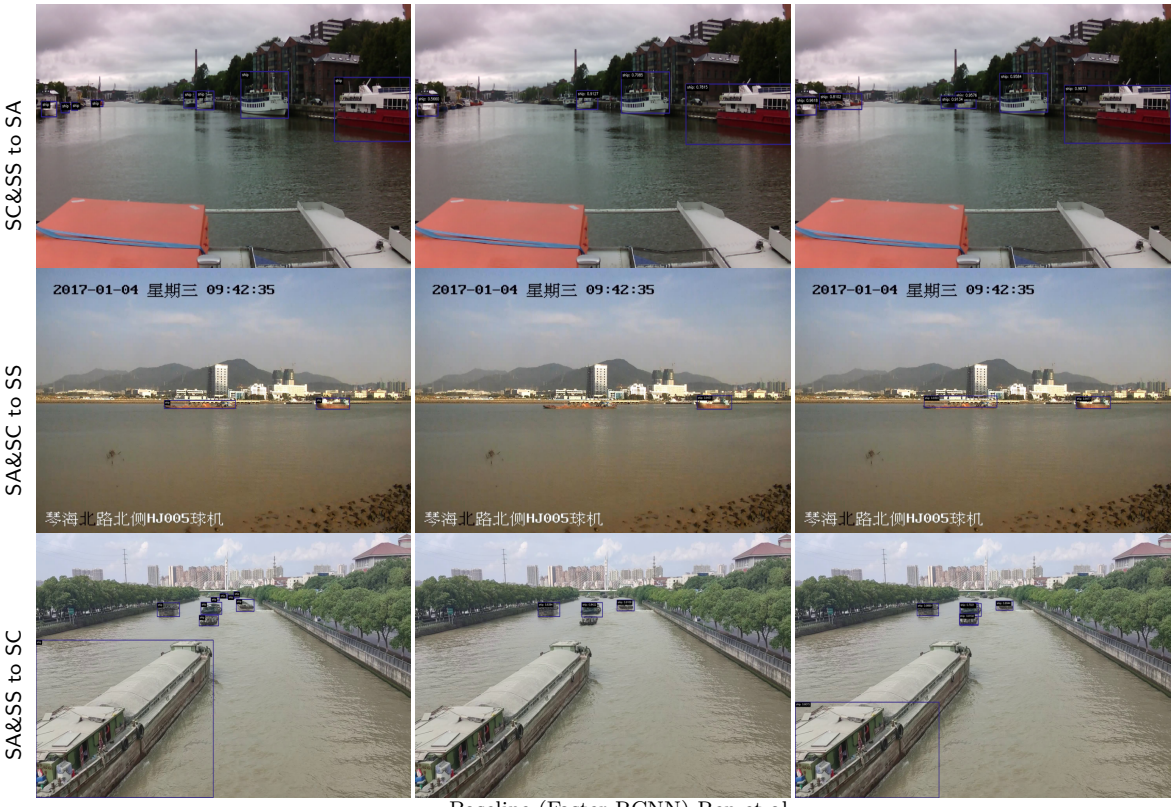
Methods	**Bike**	**Bird**	**Car**	**Cat**	**Dog**	**Person**	**mAP50**
◇Vanilla Faster RCNN Ren et al. (2015)	39.7	9.1	23.9	9.1	9.1	22.2	18.9
◇Div Danish et al. (2024)*(CVPR'24)*	41.7	12.3	29	13.2	20.6	36.5	25.5
◇DivAlign Danish et al. (2024)*(CVPR'24)*	**54.1**	16.9	30.1	**25**	**27.4**	45.9	33.2
PhysAug Xu et al. (2025b)*(AAAI'25)*	37.4	11	31.7	12.4	15.9	28.7	22.9
◇Diff. Guided (SD-1.5) He et al. (2025)*(CVPR'25)*	47.6	21	35.3	9.1	21.6	43.5	29.7
◇Diff. Guided (SD-2.1) He et al. (2025)*(CVPR'25)*	46.4	13.2	24.2	7.5	12.3	35.8	24.9
Ours.	36.4	**19**	**40.6**	23.9	22.2	**67.3**	**34.9**

Table 23: Class-wise performance of detection methods on Watercolor Real to Artistic benchmark. ◇: results are obtained directly from He et al. (2025).

Methods	**Bike**	**Bird**	**Car**	**Cat**	**Dog**	**Person**	**mAP50**
◇Vanilla Faster RCNN Ren et al. (2015)	85.7	42.5	36.4	29	18.7	54.5	44.5
◇Div Danish et al. (2024)*(CVPR'24)*	87.1	51.7	53.6	35.1	23.6	63.6	52.5
◇DivAlign Danish et al. (2024)*(CVPR'24)*	90.4	51.8	51.9	**43.9**	35.9	70.2	**57.4**
PhysAug Xu et al. (2025b)*(AAAI'25)*	73.4	47.6	56.8	25	26.7	55.8	47.6
◇Diff. Guided (SD-1.5) He et al. (2025)*(CVPR'25)*	90.1	51	48.5	40.2	28.9	66.7	54.2
◇Diff. Guided (SD-2.1) He et al. (2025)*(CVPR'25)*	**99.6**	48.4	49.1	28.4	23.4	54.2	50.6
Ours.	65.4	**57.8**	**57.3**	39.4	**45.1**	**76**	56.9

H Qualitative analysis

We present qualitative analyses for Diverse Weather, Cross Locale, Cross Maritime benchmarks. For each unseen domain, we randomly sample an image for visualization. figs. 12 to 14 show the comparison between our method, the baseline, and the ground-truth annotations. Across these examples, our approach produces more complete and accurate detections, successfully localizing objects under challenging conditions such as adverse weather and severe illumination changes. Taken together, these qualitative results further support the effectiveness of the proposed cross-domain alignment strategy in improving detection generalization across diverse unseen target domains.



Ground Truth | Baseline (Faster-RCNN) Ren et al. (2015) | **Ours.**

Figure 12: Some qualitative results on Cross Maritime benchmark.

fig. 15 shows qualitative results of different detection methods on the Diverse Weathers benchmark. Our method improved detection generalizability in various conditions, including objects undetected by baseline in heterogeneous domains.

Ground Truth | Baseline (Faster-RCNN) Ren et al. (2015) | **Ours.**

Figure 13: Some qualitative results on Cross Locale benchmark.

Ground Truth | Baseline (Faster-RCNN) Ren et al. (2015) | **Ours.**

Figure 14: Some qualitative results on Cross Locale benchmark (continue).

Ground Truth | Baseline (Faster-RCNN) Ren et al. (2015) | **Ours.**

Figure 15: Some qualitative results on Diverse weather benchmark.