# SINGLE-SOURCE GENERALIZATION FOR CROSS-GEOSPATIAL OBJECT DETECTION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Cross-geospatial shifts in remote-sensing imagery cause large drops in detector performance. These shifts are compound, as environmental conditions and scene characteristics vary together across regions. Single-source domain generalization for object detection (SDGOD) has been extensively studied under weather and illumination shifts but remains under-examined for such compound shifts. To bridge this gap, we propose CoDAlign, which combines factor-guided composite augmentation with foreground-decoupled classification alignment. First, we organize source-image augmentation around explicit geographic appearance factors and validate the induced changes in an image-anchored factor space. The same factor axes enable controlled diagnostics of the robustness gained through augmentation and the shifts that remain challenging. Second, we identify how low foreground confidence in augmented images weakens class consistency in full-distribution prediction alignment. Our decoupled alignment strengthens correct-class consistency at regions matched to ground-truth objects while retaining foreground-background and non-target class alignment. Experiments across cross-region benchmarks and multiple detector architectures demonstrate that CoDAlign consistently outperforms existing SDGOD methods.

## 1 INTRODUCTION

Object detection in remote-sensing imagery supports urban planning, environmental monitoring, disaster response, and humanitarian aid (Gui et al., 2024; Rolla et al., 2025; Quinn et al., 2018; Cheng & Han, 2016). Its success, however, rests on a training-test distribution match that breaks down when images are collected across different regions (Al-Emadi et al., 2025).

Cross-geospatial deployment exposes detectors to compound appearance shifts, as environmental conditions and scene characteristics vary together across regions. A transfer between two regions, for example from tropical Southeast Asia to the arid Middle East, entangles changes in vegetation density, soil appearance, atmospheric haze, shadow definition, and built structure. Unsupervised and source-free domain adaptation (Liu et al., 2023b; 2024c) adapt the detector with target-region imagery before deployment, and test-time adaptation (Wang et al., 2021; Ruan & Tang, 2024; Cao et al., 2026b) updates it during deployment. Both add a target-dependent adaptation step that rapid deployment may not accommodate, particularly in time-critical applications such as disaster response (Gupta et al., 2019). We therefore study this problem as single-source domain generalization for object detection (SDGOD), which trains on one labeled source region and deploys directly to unseen regions. SDGOD has an established line of methods for weather and lighting shifts in driving scenes (Wu & Deng, 2022; Danish et al., 2024).

Compound appearance shifts across geographic regions, however, pose a challenge distinct from the weather, lighting, visual-style, or synthetic-to-real shifts commonly studied in visual domain-shift research (Sakaridis et al., 2021; 2018; Muşat et al., 2021; Sakaridis et al., 2020; 2019; Loh & Chan, 2019; Inoue et al., 2018; Peng et al., 2018b;a; Chen et al., 2018), and SDGOD under such shifts remains under-examined. Existing SDGOD methods diversify the single source through image-level perturbations (Lee et al., 2024; Xu et al., 2025; Danish et al., 2024), language-guided feature transformations (Vidit et al., 2023; Zhang et al., 2025), or diffusion-generated images (Li et al., 2026; Qin et al., 2025). Yet increasing appearance diversity alone does not establish which geographic variations the augmentation covers or how detector robustness changes along them. This



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

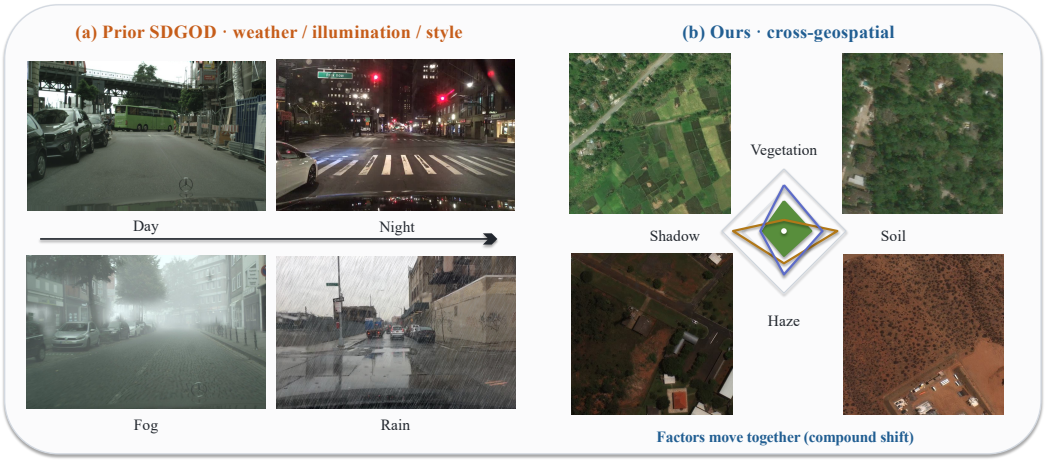



Figure 1: **A Change of Region Shifts Several Appearance Factors at Once.** (a) Prior SDGOD work studies driving scenes in which the target differs from the source by named conditions, such as day to night or clear to foggy and rainy weather. (b) Images from different regions differ simultaneously in vegetation, soil, haze, and shadow; the radar chart sketches each region as a distinct profile over these factors. CoDAlign diversifies the source along the same factor axes and uses them to diagnose which shifts remain challenging.

motivates organizing training variations and controlled robustness diagnostics around a shared set of explicit geographic appearance factors.

Appearance diversification also raises a training question: how should a detector preserve category decisions across altered views? Cross-view consistency methods answer this question by penalizing disagreement between predictions across views (Jeong et al., 2019; 2021; Lee et al., 2024; Danish et al., 2024). For the same region in the original and augmented views, the penalty is a KL-based divergence between the two predicted distributions over all classes, including background. This penalty scales class disagreement by detector's confidence that the region contains an object. When an edit makes a labeled building hard to recognize, this confidence drops, and so does the penalty for predicting a different class in the two views, even though the label says the building is there.

To address these challenges, we propose CoDAlign, a framework combining composite augmentation with decoupled classification alignment. First, we edit source images along named appearance factors, including vegetation density, soil appearance, haze, and shadows. These edited views are mixed with existing augmentations to diversify training. The same factor axes support controlled diagnostics of the robustness gained through augmentation and the variations that remain challenging. Second, our decoupled alignment strengthens correct-class consistency at regions matched to ground-truth objects while retaining foreground-background and non-target class alignment. At inference, the detector operates directly on images from unseen target regions without adaptation.

**Contributions.**

- To our knowledge, we present the first study dedicated to improving single-source domain generalization for object detection under cross-geospatial appearance shifts.
- We propose CoDAlign, which combines factor-guided composite augmentation with foreground-decoupled classification alignment to diversify training appearances and strengthen class consistency. The factor axes used for editing also support controlled diagnostics of detector robustness.
- Extensive experiments across cross-region benchmarks demonstrate that CoDAlign achieves state-of-the-art performance. Our augmentation transfers across detector architectures and consistently improves existing detection training frameworks.

## 2 Related Work

**Cross-Geospatial Generalization.** Cross-geospatial generalization has been studied in remote-sensing semantic segmentation through cross-region and cross-climate evaluation. CASID (Liu et al., 2023a) provides a climate-aware land-cover benchmark, while CrossEarth (Gong et al., 2026) and CrossEarth-Gate (Cao et al., 2026a) develop augmentation and foundation-model tuning strategies for cross-domain segmentation. These studies predict pixel-level semantics; detection requires locating and classifying individual instances amid changing geographic backgrounds. RWDS (Al-Emadi et al., 2025) pioneered benchmarking object detection under real-world geographic shifts. Building on this benchmark, we present CoDAlign in the first dedicated study, focused on improving single-source domain generalized object detection under cross-geospatial shifts.

**Domain Diversification for Single-Source Object Detection.** SDGOD methods expand source appearances through object-aware augmentation (OA-DG Lee et al., 2024), physics-based imaging degradations (PhysAug Xu et al., 2025), and Fourier corruptions (DivAlign Danish et al., 2024). PhysAug derives its degradations from atmospheric optics and simulates them by filtering the image frequency spectrum. Language-guided methods instead perturb backbone features toward text-specified appearances. CLIP the Gap (Vidit et al., 2023) learns feature offsets toward prompts describing other weather and times of day in the frozen CLIP embedding space, and SE-COT (Zhang et al., 2025) re-styles early backbone features with text-learned channel statistics, composing styles such as rain at night step by step. Generative approaches synthesize pseudo-domain images with scene- and object-level conditioning (GoDiff Li et al., 2026) or object-preserving counterfactual diffusion (OPCD Qin et al., 2025). ImageNet-E (Li et al., 2023) evaluates sensitivity to edited attributes, while CNS-Bench (Dünkel et al., 2025) studies continuous nuisance shifts and synthetic-data fine-tuning for classification. We edit source images along explicit geographic appearance factors, each checked to move its intended direction in an image embedding space more than the other directions (Section 3.1), and mix the edited views with existing augmentations. The same factor axes also serve to analyze detector robustness to appearance changes.

**Cross-View Consistency and Distillation.** Single-DGOD (Wu & Deng, 2022) uses cyclic disentanglement and self-distillation to transfer domain-invariant representations to the detector. Two of the augmentation methods above also train on each source image together with its augmented copy and penalize disagreement between the predictions on the two views. OA-DG (Lee et al., 2024) uses a JS divergence alongside object-aware contrastive learning, while DivAlign (Danish et al., 2024) uses a KL divergence between class predictions and aligns box predictions at corresponding regions. DKD (Zhao et al., 2022) makes an observation for teacher–student distillation: the standard loss couples its target-class and non-target-class terms in a way that suppresses the latter, and DKD decouples the two. We identify an analogous coupling in cross-view detection: the KL penalty discounts class disagreement at regions with low foreground confidence (Section 3.2). At regions matched to annotated objects, CoDAlign makes agreement on the labeled class, which we call correct-class consistency, independent of this confidence.

## 3 Method

**Problem setup.** Given labeled images $\mathcal{D}_s = \{(x_i, y_i)\}_{i=1}^{N}$ from one source region, where $y_i$ contains the boxes and classes in image $x_i$, we train a detector $f_\theta$ for unseen target regions without using any of their images during training.

CoDAlign combines two modules (Figure 2). Factor-guided composite augmentation mixes factor-edited images with traditional image corruptions to diversify training appearances. Foreground-decoupled classification alignment encourages the two views to agree on the class of each labeled object, regardless of the detector's confidence that the object is present. The factor axes used for editing also support post-hoc robustness diagnostics. At inference, $f_\theta$ predicts directly on unseen regions without image editing or target-domain adaptation.

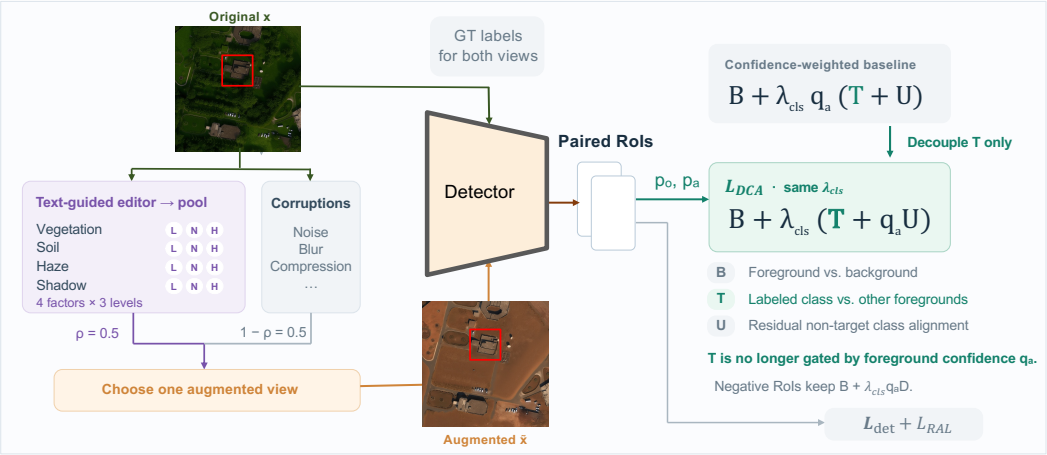



Figure 2: **CoDAlign Pipeline.** A text-guided editor changes each source image along named appearance factors, and the edits are mixed with traditional corruptions to form the augmented view. Foreground-decoupled classification alignment encourages the two views to agree on the class of each labeled object, regardless of the detector's confidence that the object is present.

### 3.1 Factor-Guided Composite Augmentation

A change of region alters how the ground and atmosphere look around the objects a detector must find. We therefore edit source images along a small set of named appearance factors: vegetation density, soil dryness and warmth, atmospheric haze, and shadow definition. A language model proposed the candidate factors, and we retained those whose edits passed the checks described below (Appendix A).

**Editing along factors.** Let $\mathcal{F}$ denote the retained factors. For each factor $j \in \mathcal{F}$ and level $\ell \in \{\text{low}, \text{neutral}, \text{high}\}$, a prompt $p_{j,\ell}$ describes the target appearance; for example, the high level of vegetation density reads "dense continuous green canopy and lush vegetation cover." A frozen text-guided image editor $G_\psi$ produces

$$x_i^{j,\ell} = G_\psi(x_i; p_{j,\ell}). \tag{1}$$

Figure 2 illustrates this editing process. Every prompt also asks the editor to keep the camera angle, object placement, and building geometry, so each edited image reuses the boxes and labels $y_i$ of its source. Four factors at three levels give up to 12 edited versions of each source image, generated once before training.

**Checking the factors.** We check each candidate factor on source images before training. Its low, neutral, and high edits must appear in this order along the factor's direction in an image embedding space (Section 3.4). Editing it must move the directions of the other factors less than its own. We also inspect object regions for destructive edits. These checks help select factors and prompts rather than filtering images; Appendix B gives the prompts, thresholds, and results.

**Mixing with traditional corruptions.** Traditional corruptions, such as noise, blur, and compression, perturb low-level image statistics that the factor edits do not target. Let $\mathcal{E}(x_i)$ be the stored edits of $x_i$, and let $\mathcal{A}$ denote a random corruption. We form the augmented view as

$$\tilde{x}_i = \begin{cases} e_i, \quad e_i \sim \text{Uniform}(\mathcal{E}(x_i)), & \text{with probability } \rho, \\ \mathcal{A}(x_i), & \text{with probability } 1-\rho, \end{cases} \tag{2}$$

with $\rho = 0.5$. Each augmented view comes from a single source, so mixing does not increase the number of training views (Appendix C).

**Use in training.** CoDAlign pairs each source image with its augmented view and aligns their predictions as described in Section 3.2. Other detection frameworks can also sample from the same edit pool while keeping their own objectives (Section 4.3).

### 3.2 FOREGROUND-DECOUPLED CLASSIFICATION ALIGNMENT

In two-view training, predictions on the original and augmented views are compared region by region. Consider a region of interest (RoI) proposed on the original image and the same region in the augmented view. For view $v \in \{o, a\}$, original and augmented, let $p_v$ be the predicted distribution over $C$ foreground classes and one background class. DivAlign (Danish et al., 2024) penalizes class disagreement with

$$\mathcal{L}_{\mathrm{KL}} = \frac{1}{M} \sum_{n=1}^{M} \mathrm{KL}(p_{a,n} \| p_{o,n}) \,, \tag{3}$$

where $M$ is the number of sampled RoIs, including those on background and the set $\mathcal{P}$ of RoIs matched to ground-truth boxes. It also aligns box predictions at the RoIs in $\mathcal{P}$ with a smooth-$L_1$ loss $\mathcal{L}_{\mathrm{RAL}}$. We keep $\mathcal{L}_{\mathrm{RAL}}$ and modify $\mathcal{L}_{\mathrm{KL}}$ as follows.

**Where the penalty weakens.** We split each prediction into two parts: the foreground confidence $q_v = 1 - p_{v,\mathrm{bg}}$, the probability that the region contains an object, and the class distribution given an object, $r_{v,k} = p_{v,k}/q_v$ for $k = 1, \ldots, C$. Writing $\mathrm{Ber}(q) = (q, 1-q)$, the chain rule of KL divergence gives

$$\mathrm{KL}(p_a \| p_o) = B + q_a D, \qquad B = \mathrm{KL}(\mathrm{Ber}(q_a) \| \mathrm{Ber}(q_o)) \,, \qquad D = \mathrm{KL}(r_a \| r_o). \tag{4}$$

Here $B$ measures disagreement on whether an object is present, and $D$ measures disagreement on its class. The class term is multiplied by the foreground confidence on the augmented view. Suppose an edit obscures a building and $q_a$ falls to 0.2; a class disagreement then costs one fifth as much as it would at full confidence.

**Using the labels.** At a background region, low foreground confidence is correct, and a small weight on class disagreement is appropriate because the region has no class. At an RoI in $\mathcal{P}$, the source labels place an object in both views. Low confidence on the augmented view then reflects a harder view rather than an empty region, and the class should still agree. For $n \in \mathcal{P}$, let $k_n$ be the class of the matched ground-truth box.

**Which part to decouple.** The class distribution can be split in the same way as the full prediction: first, how much probability goes to the labeled class $k_n$ versus all other classes together, and second, how the remaining probability spreads among those other classes. Applying the chain rule again gives $D_n = T_n + U_n$, where

$$T_n = \mathrm{KL}(\mathrm{Ber}(r_{a,n,k_n}) \| \mathrm{Ber}(r_{o,n,k_n})) \tag{5}$$

measures disagreement on the first step and $U_n$ on the second (Appendix D). The labels identify the correct class but carry no information about how probability should spread among the other classes, so we decouple only $T_n$ from foreground confidence. With two foreground classes, only one other class remains, so $U_n = 0$ and $T_n = D_n$.

**Loss.** Our loss keeps the baseline and adds a correction at labeled objects that makes the weight on $T_n$ independent of foreground confidence:

$$\mathcal{L}_{\mathrm{DCA}} = \frac{1}{M} \sum_{n=1}^{M} (B_n + \lambda_{\mathrm{cls}} q_{a,n} D_n) + \frac{\lambda_{\mathrm{cls}}}{M} \sum_{n \in \mathcal{P}} (1 - q_{a,n}) T_n. \tag{6}$$

At a labeled object, the class penalty becomes $\lambda_{\mathrm{cls}}(T_n + q_{a,n} U_n)$ instead of $\lambda_{\mathrm{cls}} q_{a,n}(T_n + U_n)$, so agreement on the labeled class no longer depends on foreground confidence. Background RoIs keep the form $B_n + \lambda_{\mathrm{cls}} q_{a,n} D_n$, which equals the baseline in Eq. 3 when $\lambda_{\mathrm{cls}} = 1$. We use $\lambda_{\mathrm{cls}} = 3$ to increase the scale; Appendix D gives the derivation, normalization, and gradients.

### 3.3 TRAINING OBJECTIVE

The detector is trained on both views with its standard detection losses $\mathcal{L}_{\mathrm{det}}$ for the region proposal network (RPN) and the RoI head. Adding the two alignment losses from Section 3.2 gives the full objective

$$\mathcal{L} = \mathcal{L}_{\mathrm{det}} + \mathcal{L}_{\mathrm{DCA}} + \mathcal{L}_{\mathrm{RAL}}, \tag{7}$$

Table 1: **Single-Source Cross-Geographic Detection on xView and xBD.** We use the CZ and HE partitions of RWDS (Al-Emadi et al., 2025) and report COCO bounding-box $AP_{50}$ (%, IoU = 0.50). Src.: source domain; OOD: mean of out-of-distribution. The final two columns report HE class-wise $AP_{50}$. Dark and light green cells mark the best and second-best values in each column. †Uses the CLIP ResNet-101 backbone of the released implementation.

| Method | Venue | **xView: CZ** (source: Tropical A) | | | | **xBD: HE** (source: Michael) | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Src. | Arid | Temp. | **OOD** | Src. | Flor. | Harv. | **OOD** | No-dmg. | Dmg. |
| FasterRCNN | NeurIPS'15 | 16.52 | 10.44 | 8.23 | 9.33 | 40.68 | 15.22 | 10.86 | 13.04 | 22.69 | 3.40 |
| Single-DGOD | CVPR'22 | 16.03 | 10.70 | 7.99 | 9.35 | 40.84 | 15.69 | 10.13 | 12.91 | 22.38 | 3.45 |
| CLIPGap† | CVPR'23 | 11.52 | 8.67 | 7.27 | 7.97 | 37.47 | 14.76 | 8.87 | 11.82 | 22.50 | 1.14 |
| DivAlign | CVPR'24 | 15.81 | 10.07 | 8.76 | 9.42 | 43.96 | 21.64 | 15.00 | 18.32 | 31.91 | 4.73 |
| PhysAug | AAAI'25 | 15.45 | 10.53 | 9.01 | 9.77 | 42.07 | 17.17 | 11.23 | 14.20 | 25.88 | 2.52 |
| GDD Student | CVPR'25 | 15.81 | 11.57 | 9.24 | 10.40 | 41.94 | 19.19 | 12.70 | 15.94 | 29.55 | 2.33 |
| FGT Student | ICCV'25 | 18.02 | 11.64 | 9.56 | 10.60 | 40.61 | 18.46 | 11.33 | 14.89 | 28.06 | 1.72 |
| Boost-VLKI | ACMMM'25 | 14.58 | 10.72 | 9.57 | 10.14 | 47.47 | 22.90 | 14.43 | 18.67 | 32.77 | 4.56 |
| LTFE† | AAAI'26 | 10.76 | 8.23 | 7.33 | 7.78 | 39.06 | 14.75 | 11.15 | 12.95 | 24.60 | 1.31 |
| **CoDAlign** | – | 16.41 | 11.89 | 9.56 | 10.72 | 45.19 | 24.65 | 17.72 | 21.19 | 35.07 | 7.30 |
| *Stable Diffusion 1.5* | | | | | | | | | | | |
| GDD Teacher | CVPR'25 | 14.16 | 10.83 | 9.04 | 9.93 | 41.38 | 25.57 | 15.56 | 20.56 | 38.61 | 2.52 |
| FGT Teacher | ICCV'25 | 15.72 | 12.06 | 10.69 | 11.38 | 40.14 | 23.48 | 14.73 | 19.11 | 36.10 | 2.12 |
| *DINOv2 / DINOv3 ViT-L* | | | | | | | | | | | |
| Cauvis | NeurIPS'25 | 13.72 | 11.62 | 10.04 | 10.83 | 42.78 | 26.75 | 19.37 | 23.06 | 37.49 | 8.64 |
| Bridge | CVPR'26 | 17.91 | 12.08 | 11.61 | 11.84 | 42.99 | 25.65 | 16.57 | 21.11 | 40.22 | 2.00 |

applied to every pair, whether the augmented view is a factor edit or a corruption. Alignment gradients flow to both views, so neither view serves as a fixed target. Appendix E specifies RoI pairing and numerical details.

### 3.4 Image-Anchored Factor Coordinates

Factor coordinates serve two purposes: they measure the checks in Section 3.1, and after training they place real source and target images on the same named axes for analysis. We anchor each factor direction in edited images rather than in text prompts, so the direction reflects the change that the editor actually produced. Let $h(x)$ be the unit-normalized feature of image $x$ from a frozen image encoder, and let $\mathcal{S}$ be a calibration subset of source training images. For factor $j$, let $\bar{h}_j^{\ell}$ be the normalized mean feature of the level-$\ell$ edits of the images in $\mathcal{S}$. The factor direction and the coordinate of an image $x$ along it are

$$d_j = \mathrm{normalize}\left(\bar{h}_j^{\mathrm{high}} - \bar{h}_j^{\mathrm{low}}\right), \tag{8}$$

$$\alpha_j(x) = h(x)^\top d_j - \mu_{s,j}, \tag{9}$$

where $\mu_{s,j}$ is the mean of $h(x_i)^\top d_j$ over the original images in $\mathcal{S}$, so the calibration images have zero mean coordinate. The directions $d_j$ form the factor bank, and $\boldsymbol{\alpha}(x)$ collects the coordinates of $x$ along all factors.

## 4 Experiments

### 4.1 Settings

**Data and splits.** We use two RWDS (Al-Emadi et al., 2025) benchmarks built on satellite imagery. Climate zones (CZ) partitions xView (Lam et al., 2018) object detection by Köppen climate zone

Table 2: **Factor Augmentation across Backbones and Detectors.** Base and +Aug report OOD COCO $AP_{50}$ (%), averaged over the two targets, before and after adding our factor-edited images; $\Delta$ is the paired change. Each pair shares the source split, initialization, training schedule, and evaluation settings while retaining the host's training objectives. Boost-VLKI and Cauvis pairs follow the native $512 \times 512$ RWDS tiles, so their Base values differ from Table 1.

| | | **xView: Climate Zones** | | | **xBD: Hurricane Events** | | |
|---|---|---|---|---|---|---|---|
| Host method | Backbone | Base | +Aug | **Δ** | Base | +Aug | **Δ** |
| | | *Faster R-CNN R50-FPN* | | | | | |
| Source-only | R50 | 9.33 | 10.08 | +0.75 | 13.04 | 16.60 | +3.56 |
| PhysAug | R50 | 9.77 | 10.39 | +0.62 | 14.20 | 16.50 | +2.30 |
| DivAlign | R50 | 9.42 | 10.18 | +0.76 | 18.03 | 19.44 | +1.41 |
| Boost-VLKI | R50 | 7.01 | 7.32 | +0.31 | 16.77 | 19.05 | +2.27 |
| | | *Faster R-CNN R50-FPN Distilled from Diffusion Teachers* | | | | | |
| GDD | R50 | 10.40 | 10.80 | +0.40 | 15.94 | 18.20 | +2.26 |
| FGT | R50 | 10.60 | 10.89 | +0.29 | 14.89 | 16.72 | +1.82 |
| | | *Grounding DINO Swin-B* | | | | | |
| Source-only | Swin-B | 16.61 | 17.12 | +0.52 | 29.97 | 31.50 | +1.53 |
| | | *DINOv2 / DINOv3 ViT-L* | | | | | |
| Cauvis | DINOv2-L | 12.29 | 12.98 | +0.69 | 19.92 | 23.35 | +3.43 |
| Bridge | DINOv3-L | 11.84 | 12.68 | +0.83 | 21.11 | 21.99 | +0.89 |

and uses 16 object classes. We train on tropical zone A and test on arid zone B and warm-temperate zone C. Hurricane events (HE) partitions the post-disaster images of xBD (Gupta et al., 2019) by event. We train on Hurricane Michael in the Florida Panhandle and test on Florence in the Carolinas and Harvey around Houston, Texas. HE derives boxes from building polygons and has two classes: no-damage, and damaged, which merges minor damage, major damage, and destroyed buildings.

**Evaluation.** Detector training and augmentation construction use only the source domain. Unless otherwise stated, we report COCO bounding-box $AP_{50}$ (%, IoU 0.50), and target-domain means weight each target equally. For repeated runs, we report the mean and standard deviation over seeds.

**Implementation.** CoDAlign builds on Faster R-CNN (Ren et al., 2015) with a ResNet-50 backbone (He et al., 2016) and feature pyramid network (Lin et al., 2017), and uses the objective in Eq. 7. Factor edits are precomputed with a frozen FLUX.1 Kontext editor (Black Forest Labs et al., 2025), and factor coordinates from a frozen RemoteCLIP ViT-L/14 encoder (Liu et al., 2024a) serve only the diagnostics. The augmentation study also uses DINO (Zhang et al., 2022) and Grounding DINO (Liu et al., 2024b), with Swin-B (Liu et al., 2021) as the latter's backbone. Pretrained backbones across the comparisons include CLIP (Radford et al., 2021), Stable Diffusion 1.5 (Rombach et al., 2022), DINOv2 (Oquab et al., 2023), and DINOv3 (Siméoni et al., 2025). Baselines follow their released configurations; each augmentation comparison keeps the host detector and training schedule fixed. Further details appear in the appendix.

### 4.2 Comparison with Existing Methods

Table 1 compares CoDAlign with source-only training and existing methods. Among methods with ResNet backbones, CoDAlign achieves the highest mean target-domain $AP_{50}$ on both benchmarks.

On climate zones, CoDAlign reaches 10.72, 1.39 points above source-only training. It exceeds the GDD and FGT students, which distill diffusion-based teachers, and Boost-VLKI, which uses vision-language knowledge. CoDAlign also attains the highest arid-zone $AP_{50}$ among ResNet-based methods, 11.89. On hurricane events, the margin is larger: CoDAlign reaches 21.19, 8.14 points

above source-only training and 2.52 points above the strongest ResNet-based baseline, Boost-VLKI. It achieves the best ResNet-based $AP_{50}$ on both Florence and Harvey.

Damaged-building detection is central to post-disaster assessment. CoDAlign more than doubles damaged $AP_{50}$ from 3.40 to 7.30, the largest gain over source-only training among methods with the same backbone. Other methods' gains on hurricane events concentrate mainly on undamaged buildings. Foreground-decoupled classification alignment (DCA) contributes directly: the matched ablation in Section 4.5 raises damaged $AP_{50}$ from 4.59 to 7.30, with consistent gains in three seeds.

Methods with large pretrained backbones serve as benchmark references, since differing pretraining data and unverified target-domain exposure limit controlled generalization comparisons (Mayilvahanan et al., 2025; Teterwak et al., 2025). On hurricane events, CoDAlign exceeds both diffusion teachers and Bridge in mean target-domain $AP_{50}$ and surpasses three of these four methods in damaged-building detection, which remains difficult despite extensive pretraining. On climate zones, Bridge, the FGT teacher, and Cauvis score higher.

### 4.3 Generalization of Factor Augmentation

Factor augmentation improves mean target-domain $AP_{50}$ in every training framework and detector we test (Table 2). We add factor-edited source images to nine hosts, each a training framework on a specific detector, spanning three detectors and four backbones. On Faster R-CNN, the hosts are Source-only, PhysAug, DivAlign, and Boost-VLKI with ResNet-50, and Bridge with a DINOv3 ViT-L backbone. GDD and FGT distill diffusion-based teacher detectors into Faster R-CNN students with ResNet-50. The remaining hosts are Source-only on Grounding DINO with Swin-B and Cauvis on DINO with a DINOv2 ViT-L backbone. Each pair retains the host's training objectives and matches the source split, initialization, training schedule, and evaluation settings. In GDD and FGT, each image in the distillation stage is replaced by one of its factor edits with probability 0.5, and the teacher and student receive the same edited image. These comparisons evaluate the augmentation module independently of DCA.

Gains range from 0.29 to 0.83 points on CZ and from 0.89 to 3.56 points on HE. On HE, Source-only improves from 13.04 to 16.60, and PhysAug and DivAlign gain 2.30 and 1.41 points. The PhysAug and DivAlign gains indicate that factor edits add training variation beyond what these existing augmentation and alignment strategies provide. The GDD and FGT students gain 0.40 and 0.29 points on CZ and 2.26 and 1.82 points on HE. Bridge, the strongest method on CZ in Table 1, gains 0.83 points on CZ and 0.89 on HE, and Cauvis gains 0.69 and 3.43 points on the two benchmarks. On Grounding DINO, factor edits raise mean target-domain $AP_{50}$ from 16.61 to 17.12 on CZ and from 29.97 to 31.50 on HE.

### 4.4 Factor-Space Diagnostics

**Factor edits cover missed shifts but push objects toward background.** For each edit, we measure how much $AP_{50}$ a model keeps relative to its $AP_{50}$ on the neutral edit (protocol in Appendix I). The hurricane targets differ from the source along all four factors, and their soil shift lies within the range of our edits (Figure 3a). Soil changes are also where existing augmentations fail. We group the high soil edits by how far they move each image; in the mildest group, the closest to the targets, Source-only, PhysAug, and DivAlign keep 5%, 14%, and 35% of their $AP_{50}$, while CoDAlign keeps 92%. Without DCA, nearly half of the edited objects receive a foreground confidence below 0.2, and these objects also show the largest class disagreement between views (Figure 3b). Full-distribution KL scales their penalty by this low confidence. DCA removes the scaling for the labeled class and reduces the disagreement by 40% on HE and 53% on CZ, most for the low-confidence objects.

**DCA helps recognize damaged buildings on real targets.** Real hurricane images lower foreground confidence even more than our edits, so the case that DCA addresses is common on real targets. On Florence, the share of damaged buildings detected with the correct class rises from 37% to 50%, and the share detected with the wrong class falls from 24% to 13%. Harvey shows the same shift, while missed buildings barely change (Figure 3c). Class confusion nevertheless remains the largest error in a TIDE analysis (Bolya et al., 2020), which makes damage recognition the main remaining challenge on these targets.

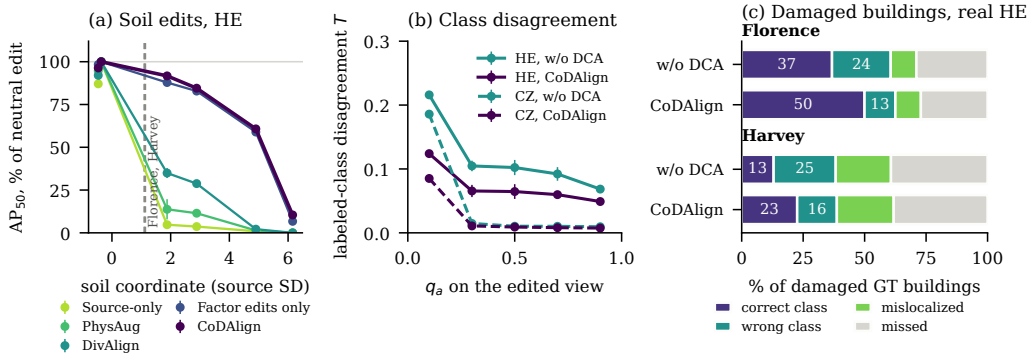



Figure 3: **Diagnostics on Held-Out Edits and Real Targets.** (a) HE $AP_{50}$ under soil edits, relative to each model's $AP_{50}$ on the neutral edit, against the soil coordinate in source-test standard deviations. High edits are binned into quartiles of the coordinate each image reaches and plotted at the bin medians; the dashed line marks the Florence and Harvey means. (b) Mean labeled-class disagreement $T$ between original and edited views, by foreground confidence $q_a$ on the edited view; w/o DCA is the $\lambda_{\text{cls}} = 3$ row of Table 3. (c) Damaged-building outcomes on the real HE targets (one-to-one matching, IoU 0.5, score threshold 0.05). All results average three seeds.

Table 3: **Ablation on He With the Michael Source.** Each row builds on the row above; the indented row replaces the change in the row above it. Src. reports $AP_{50}$ (%) on the held-out Michael split; OOD and class-wise $AP_{50}$ average Florence and Harvey. All values are means over three seeds; $\pm$ gives the OOD seed standard deviation.

| Configuration | Src. | OOD | No-dmg. | Dmg. |
|---|---|---|---|---|
| Source-only | 40.68 | 13.04 $\pm$0.76 | 22.69 | 3.40 |
| + Factor edits | 42.53 | 16.60 $\pm$0.27 | 29.46 | 3.74 |
| Generic edits instead | 41.15 | 13.97 $\pm$0.41 | 24.32 | 3.61 |
| + Corruptions | 41.77 | 17.21 $\pm$0.02 | 30.84 | 3.58 |
| + Alignment: $q_a(T+U)$ | 44.24 | 19.44 $\pm$0.31 | 35.01 | 3.88 |
| + $\lambda_{\text{cls}} = 3$: $3\,q_a(T+U)$ | 44.87 | 19.68 $\pm$0.06 | 34.76 | 4.59 |
| + DCA: $3\,(T+q_aU)$ (CoDAlign) | **45.19** | **21.19** $\pm$0.18 | **35.07** | **7.30** |

### 4.5 Ablation Studies

Table 3 adds the components of CoDAlign one at a time on hurricane events with the Michael source. Factor edits are effective on their own: randomly replacing source images by their edits during training raises OOD $AP_{50}$ from 13.04 to 16.60 in all three seeds. Generic edits from the same editor, which request a different overall visual style under the same geometry-preservation instruction, reach only 13.97. Alignment adds further robustness on two views, raising composite views from 17.21 without alignment to 19.44. DCA supplies the damaged-class gain. Raising $\lambda_{\text{cls}}$ to 3 in the confidence-weighted class term $q_a(T+U)$ changes OOD $AP_{50}$ by 0.23 points. Replacing this term with the DCA term $T + q_aU$ adds 1.51 points and raises damaged $AP_{50}$ from 4.59 to 7.30, improving both in every paired seed.

## 5 Conclusion

We presented CoDAlign for single-source cross-geospatial object detection. CoDAlign combines composite augmentation with decoupled classification alignment to expand training appearance coverage and strengthen correct-class consistency across views. Experiments across two cross-region benchmarks and multiple detector architectures demonstrate state-of-the-art performance, while our augmentation consistently improves existing detection training frameworks. Factor-level diagnostics reveal where augmentation improves robustness and highlight the remaining challenges for cross-region generalization, providing insights for future improvements.

### AI USE STATEMENT

We used GPT-5 once during method design to propose candidate visual factors; the verbatim prompt and response are reproduced in Appendix A, and every proposed candidate was then filtered by the validation in Algorithm 1 before being retained. We used GPT-5 to help with the derivation of Eq. 6. Beyond this, general-purpose AI assistants were used for light proofreading and surface-level code review. We assessed candidate edits using the source-side validation described in Appendix B. The authors take responsibility for all AI-assisted content and the final paper.

## REFERENCES

Sara A. Al-Emadi, Yin Yang, and Ferda Ofli. Benchmarking object detectors under real-world distribution shifts in satellite imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8299–8309, June 2025.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Mueller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. FLUX.1 Kontext: Flow matching for in-context image generation and editing in latent space, 2025. URL https://arxiv.org/abs/2506.15742.

Daniel Bolya, Sean Foley, James Hays, and Judy Hoffman. TIDE: A general toolbox for identifying object detection errors. In *European Conference on Computer Vision (ECCV)*, 2020.

Shilei Cao, Ziyang Gong, Hehai Lin, Yang Liu, Jiashun Cheng, Xiaoxing Hu, Haoyuan Liang, Guowen Li, Chengwei Qin, Hong Cheng, Xue Yang, Juepeng Zheng, and Haohuan Fu. CrossEarth-Gate: Fisher-guided adaptive tuning engine for efficient adaptation of cross-domain remote sensing semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13222–13233, 2026a. URL https://openaccess.thecvf.com/content/CVPR2026/html/Cao_CrossEarth-Gate_Fisher-Guided_Adaptive_Tuning_Engine_for_Efficient_Adaptation_of_Cross-Domain_CVPR_2026_paper.html.

Shilei Cao, Juepeng Zheng, Yan Liu, Baoquan Zhao, Ziqi Yuan, Weijia Li, Runmin Dong, and Haohuan Fu. Continual test-time adaptation for object detection with adaptive monitoring and randomized restoration. *arXiv preprint arXiv:2406.16439v6*, 2026b. URL https://arxiv.org/abs/2406.16439v6.

Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3339–3348, 2018.

Gong Cheng and Junwei Han. A survey on object detection in optical remote sensing images. *ISPRS journal of photogrammetry and remote sensing*, 117:11–28, 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17732–17742, 2024.

Olaf Dünkel, Artur Jesslen, Jiahao Xie, Christian Theobalt, Christian Rupprecht, and Adam Kortylewski. CNS-Bench: Benchmarking image classifier robustness under continuous nuisance shifts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19978–19988, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Dunkel_CNS-Bench_Benchmarking_Image_Classifier_Robustness_Under_Continuous_Nuisance_Shifts_ICCV_2025_paper.html.

Ziyang Gong, Zhixiang Wei, Di Wang, Xiaoxing Hu, Xianzheng Ma, Hongruixuan Chen, Yuru Jia, Yupeng Deng, Zhenming Ji, Xiangwei Zhu, Xue Yang, Naoto Yokoya, Jing Zhang, Bo Du, Junchi Yan, and Liangpei Zhang. CrossEarth: Geospatial vision foundation model for domain generalizable remote sensing semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(5):5147–5164, 2026. doi: 10.1109/TPAMI.2025.3649001.

Shengxi Gui, Shuang Song, Rongjun Qin, and Yang Tang. Remote sensing object detection in the deep learning era—a review. *Remote Sensing*, 16(2), 2024. ISSN 2072-4292. doi: 10.3390/rs16020327. URL https://www.mdpi.com/2072-4292/16/2/327.

Ritwik Gupta, Richard Hosfelt, Sandra Sajeev, Nirav Patel, Bryce Goodman, Jigar Doshi, Eric Heim, Howie Choset, and Matthew Gaston. xBD: A dataset for assessing building damage from satellite imagery. *arXiv preprint arXiv:1911.09296*, 2019. URL https://arxiv.org/abs/1911.09296.

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

Naoto Inoue, Ryosuke Furuta, Toshihiko Yamasaki, and Kiyoharu Aizawa. Cross-domain weakly-supervised object detection through progressive domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5001–5009, 2018. URL https://openaccess.thecvf.com/content_cvpr_2018/html/Inoue_Cross-Domain_Weakly-Supervised_Object_CVPR_2018_paper.html.

Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/hash/d0f4dae80c3d0277922f8371d5827292-Abstract.html.

Jisoo Jeong, Vikas Verma, Minsung Hyun, Juho Kannala, and Nojun Kwak. Interpolation-based semi-supervised learning for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11602–11611, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Jeong_Interpolation-Based_Semi-Supervised_Learning_for_Object_Detection_CVPR_2021_paper.html.

Darius Lam, Richard Kuzma, Kevin McGee, Samuel Dooley, Michael Laielli, Matthew Klaric, Yaroslav Bulatov, and Brendan McCord. xView: Objects in context in overhead imagery. *arXiv preprint arXiv:1802.07856*, 2018. URL https://arxiv.org/abs/1802.07856.

Wooju Lee, Dasol Hong, Hyungtae Lim, and Hyun Myung. Object-aware domain generalization for object detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(4):2947–2955, 2024. doi: 10.1609/aaai.v38i4.28076.

Hao Li, Xiangyuan Yang, Mengzhu Wang, Long Lan, Ke Liang, Xinwang Liu, and Kenli Li. Object style diffusion for generalized object detection in urban scene. *Pattern Recognition*, pp. 113450, 2026. doi: 10.1016/j.patcog.2026.113450.

Xiaodan Li, Yuefeng Chen, Yao Zhu, Shuhui Wang, Rong Zhang, and Hui Xue. ImageNet-E: Benchmarking neural network robustness via attribute editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20371–20381, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Li_ImageNet-E_Benchmarking_Neural_Network_Robustness_via_Attribute_Editing_CVPR_2023_paper.html.

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2117–2125, 2017.

Fan Liu, Delong Chen, Zhangqingyun Guan, Xiaocong Zhou, Jiale Zhu, Qiaolin Ye, Liyong Fu, and Jun Zhou. RemoteCLIP: A vision language foundation model for remote sensing. *IEEE Transactions on Geoscience and Remote Sensing*, 2024a. URL https://arxiv.org/abs/2306.11029.

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 *European conference on computer vision*, pp. 38–55. Springer, 2024b.

Songlin Liu, Linwei Chen, Li Zhang, Jun Hu, and Ying Fu. A large-scale climate-aware satellite image dataset for domain adaptive land-cover semantic segmentation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 205:98–114, 2023a. doi: 10.1016/j.isprsjprs.2023.09.007.

Weixing Liu, Jun Liu, and Bin Luo. Unsupervised domain adaptation for remote-sensing vehicle detection using domain-specific channel recalibration. *IEEE Geoscience and Remote Sensing Letters*, 20:1–5, 2023b.

Weixing Liu, Jun Liu, Xin Su, Han Nie, and Bin Luo. Source-free domain adaptive object detection in remote sensing images. *arXiv preprint arXiv:2401.17916*, 2024c.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin Transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10012–10022, 2021.

Yuen Peng Loh and Chee Seng Chan. Getting to know low-light images with the exclusively dark dataset. *Computer vision and image understanding*, 178:30–42, 2019.

Prasanna Mayilvahanan, Roland S. Zimmermann, Thaddäus Wiedemer, Evgenia Rusak, Attila Juhos, Matthias Bethge, and Wieland Brendel. In search of forgotten domain generalization. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/a0d2345b43e66fa946155c98899dc03b-Abstract-Conference.html.

Valentina Muşat, Ivan Fursa, Paul Newman, Fabio Cuzzolin, and Andrew Bradley. Multi-weather city: Adverse weather stacking for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pp. 2906–2915, 2021.

Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision, 2023.

Xingchao Peng, Ben Usman, Neela Kaushik, Dequan Wang, Judy Hoffman, and Kate Saenko. Visda: A synthetic-to-real benchmark for visual domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018a.

Xingchao Peng, Ben Usman, Kuniaki Saito, Neela Kaushik, Judy Hoffman, and Kate Saenko. Syn2real: A new benchmark forsynthetic-to-real visual domain adaptation. *arXiv preprint arXiv:1806.09755*, 2018b.

Hongda Qin, Xiao Lu, Zhiyong Wei, and Ningjiang Chen. Object-preserving counterfactual diffusion augmentation for single-domain generalized object detection. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 189–198, 2025. doi: 10.1145/3746027.3754959.

John A Quinn, Marguerite M Nyhan, Celia Navarro, Davide Coluccia, Lars Bromley, and Miguel Luengo-Oroz. Humanitarian applications of machine learning with remote-sensing data: review and case study in refugee settlement mapping. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 376(2128):20170363, 2018.

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 *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

Julie Rolla, Aditya Khuller, Karen An, Robert Emberson, Eric Fielding, Lori Schultz, and Kimberley Miner. Satellite-aided disaster response. *AGU Advances*, 6(1):e2024AV001395, 2025.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

Xiaoqian Ruan and Wei Tang. Fully test-time adaptation for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1038–1047, 2024.

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

Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Guided curriculum model adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. In *The IEEE International Conference on Computer Vision (ICCV)*, 2019.

Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Map-guided curriculum domain adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. doi: 10.1109/TPAMI.2020.3045882.

Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10765–10775, 2021.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3, 2025.

Piotr Teterwak, Kuniaki Saito, Theodoros Tsiligkaridis, Bryan A. Plummer, and Kate Saenko. Is large-scale pretraining the secret to good domain generalization? In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/10fc83943b4540a9524af6fc67a23fef-Abstract-Conference.html.

Vidit Vidit, Martin Engilberge, and Mathieu Salzmann. Clip the gap: A single domain generalization approach for object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3219–3229, 2023.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021.

Aming Wu and Cheng Deng. Single-domain generalized object detection in urban scene via cyclic-disentangled self-distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 847–856, 2022.

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. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(20):21815–21823, 2025. doi: 10.1609/aaai.v39i20.35487.

Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M. Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection, 2022.

Zihao Zhang, Aming Wu, and Yahong Han. Style evolving along chain-of-thought for unknown-domain object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14225–14234, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Zhang_Style_Evolving_along_Chain-of-Thought_for_Unknown-Domain_Object_Detection_CVPR_2025_paper.html.

Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11953–11962, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Zhao_Decoupled_Knowledge_Distillation_CVPR_2022_paper.html.

# A Candidate Factor Proposal Transcript

We obtained the initial candidate factor list by prompting GPT-5 with source-domain imagery and the textual constraints described in Sec. 3.1. The validation in Appendix B then filtered the proposed candidates: four factors passed all three checks and entered the retained factor bank, while the remaining candidates were dropped or revised. We reproduce the relevant exchange below.

**Prompt to GPT-5.**

> We are constructing a single-source domain generalization detector for satellite imagery in $512 \times 512$ tiles. The labeled source split comes from one geographic group, and the target groups are not seen during training. We want a small, interpretable visual factor space that can be perturbed on source tiles with a flow-matching image editor. Propose 5–8 candidate visual factors that (1) describe observable image-level appearance rather than geographic names or climate categories, (2) can be expressed as a continuous low/neutral/high axis, (3) are plausible to instantiate through prompt-guided image editing while preserving object geometry and labels, and (4) collectively cover the visual changes one might encounter between cross-geographic satellite scenes. For each factor, give a one-line definition and a one-sentence motivation.

**Response from GPT-5.**

1. **Vegetation density.** Coverage of vegetated versus bare ground, from sparse exposed soil to dense continuous canopy. Cross-geographic regions differ widely in vegetation cover.
2. **Soil dryness and warmth.** Apparent moisture and color of bare ground, from dark cool damp soils to bright warm dry soils. Soil signature is a strong regional appearance cue.
3. **Atmospheric haze veiling.** Diffuse scattering and visibility reduction across the tile, from crisp clarity to heavy haze. Atmospheric conditions vary across both season and region.
4. **Shadow definition.** Directional-lighting strength and shadow-edge sharpness, from flat diffuse lighting to crisp directional shadows. Sun angle and cloud cover differ between captures.
5. **Saturation and exposure.** Global color saturation and overall brightness of the scene, from desaturated and dim to vivid and bright. Sensor calibration and atmospheric conditions shift overall image statistics.
6. **Urban built-up density.** Fraction of impervious built surface in the tile, from sparse rural land to dense urban fabric. Built-up extent varies between geographic groups.
7. **Surface texture roughness.** Local texture frequency on ground and built surfaces, from smooth uniform regions to rough granular ones. Surface texture reflects both terrain and built-environment differences.

**Validation outcome.** We then applied the three validation checks of Appendix B to each candidate. Four factors passed and entered the retained bank used throughout the paper: vegetation density, soil dryness and warmth, haze veiling, and shadow definition. Saturation and exposure was dropped because its low-level edits produced near-black images that broke object-region label safety; an attempted prompt revision repaired pixel quality but collapsed the low/neutral/high monotonicity, so the axis was removed rather than retained as an unreliable coordinate. Urban built-up density was dropped because its edits entangled with the very object categories the detector must localize, violating the cross-factor isolation check. Surface texture roughness was folded into the appearance variation already captured by vegetation density and soil dryness, since stand-alone texture edits did not move a distinct coordinate beyond the leakage. This filtering is the empirical reason the retained factor bank is four-dimensional rather than as broad as the initial proposal list.

Table 4: **Level phrases for the retained factors.** Each phrase is inserted into the shared geometry-preserving prompt wrapper.

| Factor | Level | Level phrase |
|---|---|---|
| Vegetation density | Low | sparse scattered vegetation with broad exposed bare ground |
| | Neutral | moderate mixed vegetation coverage with visible ground gaps |
| | High | dense continuous green canopy and lush vegetation cover |
| Soil dryness and warmth | Low | darker cooler ground with muted moist-looking soil appearance |
| | Neutral | balanced earth tones with moderate soil brightness |
| | High | dry bright earth-toned soil with warm dusty ground appearance |
| Haze veiling | Low | clear crisp visibility with strong local clarity |
| | Neutral | mild atmospheric softening with moderate visibility |
| | High | diffuse veiling haze with softened contrast and reduced clarity |
| Shadow definition | Low | flat diffuse lighting with very low shadow definition |
| | Neutral | moderate shadow contrast with soft shadow edges |
| | High | strong directional sunlight with crisp shadow boundaries |

# B Factor Space Construction and Validation

Our factor space is built entirely from source-domain imagery. For RWDS-CZ, the retained factors are vegetation density, soil dryness and warmth, haze veiling, and shadow definition; each factor is instantiated through three source-side editing levels, low, neutral, and high. For a source image $x_i$ and prompt $p_{j,\ell}$, the editor produces a factor-edited view $x_i^{j,\ell}$ while the detection labels are inherited from the source image.

We use a shared prompt wrapper for all source-side factor edits:

```
Change the visual style to: [level phrase]. While
modifying the environmental style, strictly maintain the
original camera angle, object placement, and exact geometric
structure of all buildings.
```

Only the level phrase changes across factors and levels. Table 4 lists the phrases used for the four retained factors.

## B.1 Coordinate Construction

We use the frozen RemoteCLIP ViT-L/14 image encoder (Liu et al., 2024a) with its OpenCLIP evaluation preprocessing. Each RGB image is encoded independently, and its feature is normalized to unit Euclidean norm. For a calibration set $\mathcal{S}$, Eq. 8 first averages the features within each factor–level cell and then normalizes each mean. The normalized high-minus-low difference defines the direction $d_j$; neutral edits are reserved for checking the ordering along this direction.

Let $K = |\mathcal{F}|$ and let $d$ be the image-feature dimension. Stacking the directions as rows gives $\boldsymbol{D} \in \mathbb{R}^{K\times d}$. With column-vector features, the coordinate computation is

$$\boldsymbol{\alpha}(x) = \boldsymbol{D}h(x) - \boldsymbol{\mu}_s, \qquad \boldsymbol{\mu}_s = \frac{1}{|\mathcal{S}|}\sum_{x_i\in\mathcal{S}} \boldsymbol{D}h(x_i). \tag{10}$$

The center uses original source images, so their mean coordinate is zero. A neutral edit need not have zero coordinate. We retain the individual factor directions without orthogonalization or variance normalization, preserving their correspondence to the editing prompts.

## B.2 Candidate Validation and Interpretation

For edit family $j$, level $\ell$, and coordinate $k$, define the mean projected coordinate

$$m_{j\ell,k} = \frac{1}{|\mathcal{S}|}\sum_{x_i\in\mathcal{S}} \alpha_k(x_i^{j,\ell}). \tag{11}$$

**Algorithm 1** Source-side candidate factor validation.

**Require:** Candidate factors $\mathcal{F}$, source calibration set $\mathcal{S}$, fixed editor $G_\psi$, frozen encoder $h$, leakage threshold $\tau = 0.7$
**Ensure:** Retained factor bank $\boldsymbol{D}$
**for** each candidate $j \in \mathcal{F}$ **do**
  Generate low, neutral, and high edits for images in $\mathcal{S}$
  Construct $d_j$ using Eq. 8
**end for**
Stack candidate directions and compute source-centered coordinates
**for** each candidate $j \in \mathcal{F}$ **do**
  Check low/neutral/high ordering using Eq. 11
  Check the on-target span and off-target leakage using Eq. 12
  Inspect scene layout and annotated object regions for destructive edits
  Retain passing candidates; revise failing prompts or remove the candidate
**end for**
If prompts change, regenerate affected edits and repeat construction and validation
**return** the retained directions as $\boldsymbol{D}$

The ordering check requires $m_{j\mathrm{low},j} < m_{j\mathrm{neutral},j} < m_{j\mathrm{high},j}$. Since the endpoints also construct the axis, this is an internal consistency check of the edits and encoder. For a nonzero on-target span, cross-factor leakage is

$$L_{jk} = \frac{|m_{j\mathrm{high},k} - m_{j\mathrm{low},k}|}{|m_{j\mathrm{high},j} - m_{j\mathrm{low},j}|}. \tag{12}$$

Thus $L_{jj} = 1$, and off-diagonal entries measure how strongly editing factor $j$ moves coordinate $k$ relative to its intended coordinate. The candidate-validation threshold is $\max_{k \neq j} L_{jk} \leq 0.7$. A zero on-target span fails validation.

We also inspect scene layout and annotated object regions, using source–edit image similarity and ground-truth-region SSIM to identify destructive edits. These checks guide candidate and prompt revision; they do not constitute per-image acceptance filtering of the full training pool. Generated views inherit source annotations, so residual editing artifacts can still introduce supervision noise. Figures 4 and 5 show the existing leakage and object-region diagnostics.

For post-hoc analysis, we keep the encoder, bank, and source center fixed when projecting real-domain images and synthetic edits. Coordinate distributions summarize appearance shifts within this shared representation. Detector responses to edits measure sensitivity to the corresponding synthetic interventions; they do not assign causal shares of a real-domain performance gap to individual factors.

# C Composite Augmentation Details

## C.1 Calibration Images and Edit Pools

The calibration set $\mathcal{S}$ is a subset of the labeled source training split used to construct factor coordinates and inspect candidate edits. The detector's training set is $\mathcal{D}_s$; coordinate calibration does not restrict detector training to $\mathcal{S}$. For each source image, generated files are indexed by source identifier, factor, and editing level, preserving their link to the original annotations. A complete pool with $K$ retained factors and three levels contains $3K$ alternatives per source image. With the four retained factors, this gives 12 alternatives; the sampler draws from the available entries in $\mathcal{E}(x_i)$. The editor and diagnostic encoder remain fixed throughout this process.

## C.2 Sampling and Label Handling

At each visit to a source sample, Eq. 2 selects the factor-edit branch with probability $\rho = 0.5$. That branch samples uniformly over the image's available factor–level variants. The traditional-corruption branch samples one corruption type uniformly and one severity uniformly from levels 1 through 5. Our corruption set contains defocus, glass, Gaussian, and motion blur; speckle, shot,

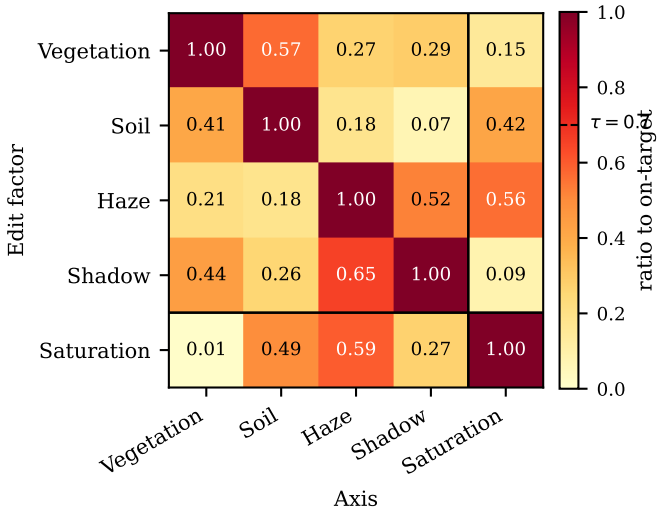



Figure 4: **Cross-factor leakage matrix.** Off-target high−low span (column axis) divided by the on-target span of the row edit; diagonal = 1.

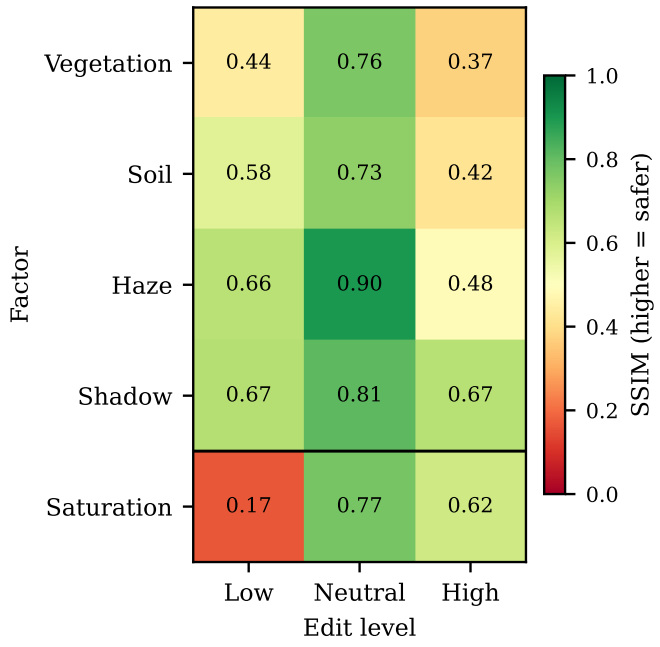



Figure 5: **Object-region label safety (GT-ROI SSIM).** SSIM between source images and factor edits over ground-truth boxes. The saturation/low cell collapses to 0.168 as the low prompt drives images near-black, breaking object regions; this failure, not leakage, drops saturation before training.

impulse, and Gaussian noise; JPEG compression; pixelation; elastic transformation; brightness, saturation, and contrast changes; high-pass filtering; and Fourier phase scaling. These operations follow the corruption implementation in the two-view training framework (Danish et al., 2024).

A training sample contains the original image and one augmented image, regardless of which branch is selected. Factor edits are sampled as alternative appearances of the source image, so storing more variants does not increase the number of training samples per epoch. Both branches retain the source

class labels and boxes. The paired images share resizing and random flipping, and the boxes undergo the corresponding spatial transformation. For a stored edit, the loader matches the original view's resized shape and applies its recorded flip. This provides a shared coordinate system for detection supervision and RoI alignment; the object-preservation checks address distortions introduced during generation.

# D Decoupled Classification Alignment: Derivation

## D.1 Foreground–Background Decomposition

For one paired RoI, suppress its index and use the notation of Sec. 3.2. The class distribution is $p_v = (q_v r_{v,1}, \ldots, q_v r_{v,C}, 1 - q_v)$ for $v \in \{o, a\}$. Expanding the augmented-to-original KL gives

$$\begin{aligned} \mathrm{KL}(p_a \| p_o) &= \sum_{k=1}^{C} q_a r_{a,k} \log \frac{q_a r_{a,k}}{q_o r_{o,k}} + (1 - q_a) \log \frac{1 - q_a}{1 - q_o} \\ &= q_a \log \frac{q_a}{q_o} + (1 - q_a) \log \frac{1 - q_a}{1 - q_o} + q_a \sum_{k=1}^{C} r_{a,k} \log \frac{r_{a,k}}{r_{o,k}} \\ &= B + q_a D. \end{aligned} \tag{13}$$

The second line uses $\sum_k r_{a,k} = 1$. The coefficient of $D$ is the augmented foreground probability because the expectation in this KL direction is taken under $p_a$.

## D.2 Assigned Class Versus Other Foreground Classes

For a positive RoI $n$ with assigned ground-truth class $k_n$, let $t_{v,n} = r_{v,n,k_n}$. Define the distribution over the remaining foreground classes by

$$s_{v,n,k} = \frac{r_{v,n,k}}{1 - t_{v,n}}, \qquad k \neq k_n. \tag{14}$$

Applying the same chain rule within the foreground distribution yields

$$D_n = T_n + U_n, \qquad U_n = (1 - t_{a,n}) \mathrm{KL}(s_{a,n} \| s_{o,n}). \tag{15}$$

Here $T_n$ compares the assigned class against the sum of the other foreground classes, while $U_n$ compares their relative probabilities. The background class is excluded from both groups and is handled by $B_n$. Adding the correction in Eq. 6 gives

$$q_{a,n} D_n + (1 - q_{a,n}) T_n = T_n + q_{a,n} U_n. \tag{16}$$

Thus the positive-RoI penalty is $B_n + \lambda_{\mathrm{cls}}(T_n + q_{a,n} U_n)$. Background-negative RoIs keep $B_n + \lambda_{\mathrm{cls}} q_{a,n} D_n$. For two foreground classes, $s_v$ has a single entry, so $U_n = 0$ and $T_n = D_n$. For more classes, the correction strengthens the assigned-class distinction while preserving confidence weighting among the other classes.

## D.3 Normalization and Gradients

Both sums in Eq. 6 use the total sampled original-view RoI count $M$. The correction therefore scales with the fraction of positive RoIs in the batch; it is not renormalized by $|\mathcal{P}|$. If $\mathcal{P}$ is empty, its contribution is zero and background-negative alignment remains active. If there are no sampled RoIs, the implementation returns zero alignment losses.

Both views receive gradients through their predictions, including $q_a$. Holding the conditional distributions fixed, the positive class penalty has derivative $\lambda_{\mathrm{cls}} U$ with respect to $q_a$, compared with $\lambda_{\mathrm{cls}}(T + U)$ before correction. This describes the explicit dependence of the loss on foreground confidence; the detector logits jointly determine all terms during optimization.

# E Two-View Training Implementation

## E.1 Corresponding Regions and Detection Supervision

The shared detector processes both views and computes its standard supervised RPN and RoI losses with the inherited annotations. For the alignment losses, we use the original view's sampled proposals and their ground-truth assignments. The same RoI coordinates are pooled from the original and augmented feature maps, so each aligned pair refers to one source region. Positive indices and assigned labels come from the original-view matcher and sampler. The augmented view still contributes its own supervised detection losses through the detector's normal training path.

## E.2 Stable Classification Loss Evaluation

Let $z_{v,n,k}$ be a foreground-class logit for view $v$ and RoI $n$. We evaluate the added target-versus-rest term using two grouped logits,

$$g_{v,n} = \left( z_{v,n,k_n},\ \log \sum_{k \neq k_n} \exp z_{v,n,k} \right). \tag{17}$$

A log-softmax over $g_{v,n}$ gives the log probabilities of the assigned foreground class and its complement. The second component uses log-sum-exp over the other foreground logits, excluding background. This computes $T_n$ without explicitly dividing by small foreground or complementary probabilities. The correction uses $q_{a,n}$ computed by summing foreground probabilities from the full softmax.

The retained foreground–background decomposition uses log-softmax and clamps background probabilities to $[\epsilon, 1-\epsilon]$, with $\epsilon = 10^{-6}$, before evaluating logarithms. The algebraic identities above describe normalized probabilities; numerical clipping can change their values at saturated predictions. We set $\lambda_{\text{cls}} = 3$.

## E.3 Localization Alignment and Inference

Let $b_{v,n}$ be the four predicted box offsets for the assigned class at positive RoI $n$. Writing $\text{smooth}_{L_1}(u;\beta)$ for the smooth-$L_1$ function with transition $\beta$, localization alignment is

$$\mathcal{L}_{\text{RAL}} = \frac{1}{4|\mathcal{P}|} \sum_{n \in \mathcal{P}} \sum_{d=1}^{4} \text{smooth}_{L_1}(b_{a,n,d} - b_{o,n,d}; 1/9). \tag{18}$$

This term is zero when $\mathcal{P}$ is empty, and both box predictions receive gradients. At inference, the detector processes each image through its standard prediction pipeline. Image editing, factor-coordinate computation, and paired-view alignment are used during data preparation, analysis, or training and add no inference parameters.

# F Licenses and Intended Use of Artifacts

We list the data, code, and model artifacts used in this work, together with their licenses and intended use. All artifacts are used for non-commercial academic research, consistent with their stated terms.

**Datasets.** RWDS-CZ Al-Emadi et al. (2025) is released under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0) license; we use it for research only and redistribute no derivatives outside this academic context.

**Code frameworks.** MMDetection is released under the Apache License 2.0. The FLUX.1 Kontext reference implementation is released under the Apache License 2.0. Both are used for the research purposes for which they are distributed.

Table 5: **Image and annotation counts per RWDS-CZ split.**

| Split | CZ A | | CZ B | | CZ C | |
|---|---|---|---|---|---|---|
| | Img | Ann | Img | Ann | Img | Ann |
| Train | 8,474 | 208,397 | 6,456 | 76,599 | 9,230 | 229,629 |
| Val | 3,698 | 104,876 | 2,871 | 24,470 | 4,169 | 84,208 |
| Test | 3,861 | 100,025 | 2,753 | 43,669 | 4,294 | 109,266 |

**Pre-trained models.** The Faster R-CNN R50-FPN and Grounding DINO Swin-B backbones are MS-COCO pre-trained checkpoints from the MMDetection model zoo and inherit its Apache 2.0 terms. RemoteCLIP Liu et al. (2024a) is used as the frozen image encoder for factor coordinate computation; we follow its released weights and intended research use. The pre-trained FLUX.1 Kontext editor weights from Black Forest Labs are used as released, for non-commercial research.

**Baselines.** Single-DGOD, CLIP-Gap, DivAlign, LTFE, and PhysAug results were obtained by training the authors' released implementations on the RWDS-CZ dataset and protocol. Single-DGOD and DivAlign are MIT licensed; PhysAug is Apache 2.0 licensed; CLIP-Gap and LTFE follow the use terms stated by their authors.

## G Dataset Documentation and Statistics

**Overview.** RWDS-CZ provides $512 \times 512$ aerial RGB patches with COCO-format bounding boxes over 16 object categories, grouped into three climate domains: CZ A (tropical), CZ B (arid), and CZ C (temperate). Each domain comes with disjoint train, validation, and test splits.

**Object categories.** The 16 categories are Passenger Vehicle, Small Car, Bus, Utility Truck, Truck, Cargo Truck, Truck Tractor with Box Trailer, Trailer, Cargo/Container Car, Dump Truck, Front Loader/Bulldozer, Shed, Building, Vehicle Lot, Shipping Container Lot, and Shipping Container.

**Per-split statistics.** Table 5 reports image and annotation counts for each split of each climate domain, as parsed from the released COCO JSON files. Objects are small relative to the image canvas: across all categories and domains, the median annotation box has a long-side extent of roughly 34 pixels in a $512 \times 512$ tile.

**Source calibration subset.** The factor-coordinate calibration subset is drawn from the source training split, as described in Appendix C.1. Evaluation uses the full release test split of each target domain.

## H Compute Budget and Implementation Details

**Hardware and software.** Experiments use local NVIDIA L40 GPUs and cluster GPUs, with software environments matched to each implementation. CoDAlign uses MMDetection v3; comparison methods retain their required framework versions. Factor-edited views are precomputed offline by the image editor; detector training reads them as standard image files.

**Training configurations.** CoDAlign uses the Faster R-CNN R50-FPN MixDiv recipe with corrected RoI pairing, retaining the detection and localization objectives while adding the correct-class alignment increment. The CZ and HE runs use the existing 12-epoch host schedule, with three repeats of the source training dataset per epoch. For the primary baseline comparison, Boost-VLKI builds on its released Cityscapes-C configuration, while Bridge builds on its released DroneVehicle configuration with a DINOv3 ViT-L backbone. We rerun these methods and CLIPGap/LTFE with training short sides sampled from 640–800 pixels and a test short side of 800, with a maximum side of 1333. Boost-VLKI uses two epochs with eight dataset repeats and total batch size eight; Bridge uses 12,000 iterations and total batch size 16. CLIPGap and LTFE keep the CLIP ResNet-101 backbone and initialization of their released implementations, with 100,000 iterations and total batch

size four; CLIPGap spends the first 1,000 iterations optimizing semantic offsets before training the detector. Cauvis uses 12 epochs and total batch size 16, with the two training geometries compared below. These reruns use seed zero and one RTX PRO 6000 Blackwell GPU per model, retaining method-specific initialization and training objectives. GDD and FGT retain their existing benchmark adaptations, with a test short side of 800 and maximum side of 1333. The paired augmentation experiments use a separate protocol that fixes these settings within each host-method pair.

**Method-specific computation.** Composite Augmentation stores factor edits offline and samples one augmented view per source image during training. DCA reuses the detector's class predictions at corresponding RoIs and introduces no trainable parameters. The diagnostic encoder is separate from the detector and is used to construct and score factor coordinates. Loss evaluation and normalization are specified in Appendix E.

**Reported numbers.** CoDAlign results use seeds 0, 1, and 2. Comparison results include both single-seed and multi-seed runs; the seed count accompanies each result, with means and standard deviations reported for multiple runs. Augmentation gains are computed from matched runs with the same seed and host training settings.

### H.1 Cauvis Training Geometry

We compare two Cauvis training geometries with the same initialization, seed, batch size, and 12-epoch schedule. The range variant samples a short side from 640–800 pixels without random cropping. The DINO variant uses eleven scales from 480 to 800 in increments of 32 and a random-crop branch. Both variants use a test short side of 800 and maximum side of 1333; Table 1 reports the DINO variant. This comparison changes both the scale distribution and cropping, so the difference does not isolate either component.

Table 6: Cauvis training geometry on CZ A and HE Michael (seed 0). Values are COCO bbox mAP (%); OOD averages B/C for CZ and Florence/Harvey for HE.

| | CZ | | | | HE | | | |
|---|---|---|---|---|---|---|---|---|
| Training geometry | A | B | C | OOD | Michael | Florence | Harvey | OOD |
| Range 640–800 | 2.76 | 1.96 | 2.21 | 2.08 | 12.59 | 7.57 | 5.88 | 6.73 |
| DINO scales + crop | 5.82 | 5.09 | 4.00 | 4.54 | 18.64 | 13.34 | 7.48 | 10.41 |

## I Diagnostic Protocol

The diagnostics in Section 4.4 use 150 validation images per source region, taken from scenes not used in training. Each image is edited at every factor level with the training prompts and corrupted with three random draws of the training corruptions. For class consistency, we pass each ground-truth box as an RoI through the original and edited images and compute $T$ from Eq. 5.

## J DCA Ablation on CZ

On CZ, Table 7 isolates DCA with composite augmentation and $\lambda_{\text{cls}} = 3$ held fixed. DCA raises mean OOD $AP_{50}$ from 10.18 to 10.72, with positive changes in two of three seeds. The improvement is concentrated in B, while C and source $AP_{50}$ decrease slightly.

Table 7: **DCA ablation on CZ with the A source.** Both rows use the same composite augmentation and $\lambda_{\text{cls}} = 3$; the formulas show the positive-RoI class term. Src., B, and C report $AP_{50}$ (%) on their test splits; OOD averages B and C. Values are means over three seeds, with the OOD seed standard deviation.

| Configuration | Src. | B | C | OOD |
|---|---|---|---|---|
| Without DCA: $3\,q_a(T+U)$ | 16.10 | 10.99 | 9.36 | 10.18 $\pm$0.26 |
| With DCA: $3\,(T+q_aU)$ | 16.41 | 11.89 | 9.56 | 10.72 $\pm$0.04 |