Beyond Photorealism: Counterfactual Synthetic Bundles for Invariant Sim-to-Real Vision

Anonymous CVPR submission

Paper ID ****

Abstract

Synthetic data is often treated as a scalable source of labeled images, with most prior work improving the realism, controllability, or utility of individual samples. We argue that its main advantage is different: synthetic engines can generate matched interventions on the same scene while preserving labels and factor metadata. We introduce Counterfactual Synthetic Bundles *(CSBs), which expand each anchor scene into a small set of one-factor interventions, and* CaBRA *(**C**ounterfactual **B**undle **R**epresentation **A**lignment), a bundle-aware vision architecture that separates task-relevant semantics from nuisance factors through consensus distillation, intervention supervision, and leakage suppression. CaBRA enforces stable predictions across label-preserving shifts such as lighting, weather, compression, and sensor noise, while handling geometry-changing interventions through equivariant consistency. We also introduce* Bundle Stability Score*,* Intervention Leakage*, and* Counterfactual Transfer Correlation *to measure whether synthetic data teaches useful invariances and whether they predict real-world transfer. Our central hypothesis is that, under a fixed render budget, structured counterfactual bundles improve sim-to-real accuracy, robustness, calibration, and sample efficiency over i.i.d. synthetic images.*

1. Introduction

Synthetic data has become increasingly important in computer vision because it offers scalable labels, precise annotations, and strong control over scene content. Recent work has improved the realism, controllability, and downstream utility of synthetic samples through layout-controlled generation, task-aware generation, and large-scale synthetic pretraining pipelines [3, 8, 10]. At the same time, the central question remains open: *what is the right way for a vision model to use synthetic data?*

Most current pipelines treat synthetic images as independent labeled examples. Even when a simulator or structured generator knows exactly which factor changed across renders—for example lighting, weather, compression, background style, or camera pose—that information is usually discarded after rendering. The learner is trained on the final images and labels alone, as if the samples had been drawn i.i.d. from a generic source distribution. This ignores the strongest property of synthetic data: the ability to generate *matched interventions* on the same underlying scene.

We argue that synthetic data should not primarily be viewed as a cheaper replacement for real data, but as a source of *supervised counterfactual structure*. A rendering engine can preserve semantic scene identity while varying only one nuisance factor at a time. That means synthetic data can tell the learner not only *what* the label is, but also *what should remain unchanged* when lighting changes, or *what should remain consistent up to a known transform* when the camera moves. Ordinary domain randomization and standard augmentations do not provide this kind of structured supervision.

This observation matters because synthetic-data evaluation remains incomplete. Recent work has argued that the domain gap between synthetic and real data is still difficult to measure directly [2], and synthetic-only models can remain brittle under corruption and real-world shifts despite strong source-domain performance [9]. These failures suggest that realism alone is not enough. A synthetic pipeline should also be judged by whether it teaches the correct invariances.

In this paper, we propose *Counterfactual Synthetic Bundles* (CSBs), where each anchor scene is expanded into a small set of intervention-linked variants that differ in exactly one factor family at a time. We pair CSBs with *CaBRA*, a bundle-aware architecture that learns two complementary representations: a semantic branch that should support the downstream task while rejecting nuisance variation, and a nuisance branch that should explicitly encode factor identity and intervention magnitude. The semantic branch is supervised with bundle consensus distillation, invariant consistency over label-preserving interventions, and equivariant consistency over geometry-changing interventions when exact transforms are known.

Our formulation differs from prior synthetic-data work in two important ways. First, our objective is not to improve the utility of *individual* generated samples, but to exploit *relationships among generated samples*. Second, we contribute not only a learner, but also bundle-native evaluation metrics that quantify whether a synthetic pipeline truly teaches invariance. We show how to construct, train on, and evaluate counterfactual bundles under a fixed render budget.

Our contributions are fourfold:

1. We introduce **Counterfactual Synthetic Bundles** (CSBs), a synthetic-data formulation in which each anchor scene is rendered into sparse one-factor interventions that preserve exact labels and factor metadata.
2. We propose **CaBRA**, a bundle-aware architecture that factorizes task-relevant semantics from nuisance factors using consensus distillation, intervention supervision, adversarial leakage suppression, and optional geometric alignment.
3. We define **bundle-native metrics**—Bundle Stability Score, Intervention Leakage, and Counterfactual Transfer Correlation—to measure whether synthetic data teaches the intended invariances and whether these predict real-world transfer.
4. We specify an **adaptive factor scheduler** that allocates render budget toward nuisance families to which the current model is most fragile.

The core thesis of this work is simple: the unique value of synthetic data is not merely that it is labeled and scalable, but that it can provide supervised interventions. If the learner is redesigned to exploit that structure, synthetic data can be useful for reasons that extend beyond photorealism.

2. Related Work

2.1. Synthetic data for computer vision

Synthetic data is widely used in vision because it provides scalable labels, precise annotations, and direct control over scene factors. Recent work has focused mainly on improving the quality, controllability, or downstream utility of individual synthetic samples. AeroGen uses layout-controlled diffusion for remote-sensing detection [10], UtilGen optimizes generation using downstream utility feedback [3], and DAViD shows that high-quality synthetic data can support strong dense-prediction models [8]. Our work is complementary but shifts attention from individual samples to *matched intervention bundles*.

2.2. Domain randomization and sim-to-real transfer

A large sim-to-real literature studies domain randomization, style transfer, and synthetic pretraining. The usual goal is to increase source diversity so models rely less on simulator-specific cues. However, standard randomization often changes many factors at once, which weakens supervision about what should stay invariant. We instead use sparse one-factor bundles, which make the intended invariances explicit.

2.3. Causal and counterfactual learning

Recent work suggests that counterfactual or causal supervision can improve robustness and transfer. Sim-to-Real Causal Transfer studies causal regularization for trajectory prediction [4], CounterPC uses counterfactual feature realignment for point-cloud adaptation [12], and Counterfactual Contrastive Learning shows benefits from explicit real-counterfactual pairing [6]. Synthetic Counterfactual Faces uses targeted synthetic perturbations to probe model sensitivity [5]. Our work extends this idea to generic synthetic-to-real vision by combining matched bundles, semantic–nuisance factorization, and bundle-aware evaluation.

2.4. Evaluating synthetic data

Evaluating synthetic data remains difficult. DoGSS-PCL proposes a metric for measuring the real–synthetic domain gap in point clouds [2], while robustness studies show that synthetic-only models can still be brittle under corruption and adversarial noise [9]. We address this gap with bundle-native metrics that measure whether synthetic interventions teach the invariances needed for real-world transfer.

2.5. Positioning of this work

Our method differs from prior work in three ways. First, we treat synthetic generation as a way to create *supervised interventions*, not just more labeled data. Second, we design the learner to exploit matched interventions directly through bundle-aware training. Third, we evaluate synthetic data using metrics tied to intervention stability rather than realism alone. This makes our approach an intervention-aware representation learning framework for sim-to-real vision.

3. Method

3.1. Overview

We propose *Counterfactual Synthetic Bundles* (CSBs) and a bundle-aware learner called *CaBRA*. The central idea is to replace independent synthetic samples with matched bundles of sparse one-factor interventions and to train a model that explicitly separates task-relevant semantics from nuisance variation. CaBRA is a training-time bundle-aware architecture and a test-time single-image predictor.

3.2. Problem formulation

Let a scene be described by latent variables

$$z = (c, n),$$

where c denotes task-relevant content and $n = \{n_1, \ldots, n_K\}$ denotes nuisance factors. Content includes object identity, geometry, layout, and semantic structure. Nuisance factors include lighting, weather, sensor corruption, texture style, background context, and viewpoint.

An image is rendered by a simulator or generator

$$x = G(c, n),$$

and the task label is given by

$$y = f(c).$$

Standard synthetic training samples independent pairs (x, y). Instead, we define a *Counterfactual Synthetic Bundle*

$$B = \{(x_0, y_0, m_0), \ldots, (x_M, y_M, m_M)\},$$

where all bundle members share the same underlying content c, and each non-anchor member is produced by a sparse intervention on exactly one factor family:

$$x_j = G(c, do(n_{k_j} = v_j), n_{-k_j}).$$

Each metadata record m_j contains factor family, intervention magnitude, visibility masks, and, when available, exact alignment transforms.

3.3. Invariant and equivariant interventions

We split interventions into two sets.

Invariant interventions. These include lighting, exposure, weather, compression, blur, noise, color shifts, texture style, and background style. They are intended to preserve the semantic label:

$$y_j = y_0.$$

Equivariant interventions. These include viewpoint or camera changes that alter the pixel-space arrangement of the label while preserving scene identity. In this case,

$$y_j = T_j(y_0),$$

where T_j is the known transformation induced by the intervention.

This distinction is important. For label-preserving changes we want invariance; for geometry-changing interventions we want consistency after alignment rather than raw invariance.

3.4. Counterfactual Synthetic Bundle generation

Each bundle begins from an anchor scene

$$s_a = \{\text{obj}, \text{layot.}, \text{mtrls.}, \text{t}, \text{weathr}, \text{snsr}, \text{background}\}.$$

We render one anchor image x_0 and then generate M sparse interventions by changing exactly one factor family at a time.

For the default configuration, each bundle contains five images:

1. Anchor scene
2. Illumination / exposure intervention
3. Weather intervention
4. Sensor / corruption intervention
5. Context / occlusion / style intervention

Every fourth bundle is replaced by a geometry-changing bundle when exact alignment metadata is available.

Bundle validity. Not every generated variant is a valid counterfactual. We therefore accept a candidate invariant intervention only if:

1. scene graph topology is unchanged,
2. object identities are unchanged,
3. semantic mask IoU with the anchor exceeds 0.98,
4. normalized depth MAE is below 0.005,
5. camera parameters are unchanged.

For equivariant interventions we require visible aligned overlap greater than 0.6 and warped-label IoU greater than 0.97.

3.5. Adaptive factor scheduler

Different nuisance families do not deserve equal render budget. Some are already handled well by the current model; others still cause severe prediction instability. We define a factor fragility score

$$\text{Frag}_k = \mathbb{E}_B \left[\ell_{\text{task}}(x_k) - \ell_{\text{task}}(x_0) + \eta D_{\text{pred}}(\hat{y}_k, \hat{y}_0)\right].$$

We then sample future intervention families using

$$q(k) \propto \exp\left(\alpha\, \text{Frag}_k + \beta\, \text{Rarity}_k - \gamma\, \text{Cost}_k\right).$$

In all experiments, we use $\eta = 0.5$, $\alpha = 1.0$, $\beta = 0.25$, and $\gamma = 0.15$, and update $q(k)$ every 2,000 training steps with EMA momentum 0.9.

3.6. CaBRA architecture

Each image x_j in a bundle is passed through a shared encoder E_θ:

$$F_j = E_\theta(x_j).$$

For the primary segmentation experiments we instantiate E_θ as SegFormer-B5 [11], though the framework is backbone-agnostic.

If the intervention is geometry-preserving, alignment is identity. If the intervention changes viewpoint, we align the features to anchor coordinates using exact simulator metadata:

$$F_j^{\text{align}} = \text{Align}(F_j, A_j).$$

We then project aligned features into two subspaces:

$$Z_j^s = P_s(F_j^{\text{align}}), \qquad Z_j^n = P_n(F_j^{\text{align}}),$$

where Z_j^s is the semantic representation and Z_j^n is the nuisance representation.

Semantic branch. The semantic branch should support the downstream task while being invariant to nuisance interventions.

Nuisance branch. The nuisance branch should encode factor family, intervention magnitude, and severity.

3.7. Bundle consensus distillation

Given a bundle, we compute a consensus semantic representation

$$\bar{Z}^s = \mathrm{Agg}(Z_0^s, \ldots, Z_M^s),$$

using a lightweight multi-head attention aggregator. This consensus representation acts as a teacher for each view-specific semantic representation:

$$\mathcal{L}_{\mathrm{distill}} = \frac{1}{M+1} \sum_{j=0}^{M} \left\| Z_j^s - \mathrm{sg}(\bar{Z}^s) \right\|_2^2,$$

where $\mathrm{sg}(\cdot)$ denotes stop-gradient.

The intuition is that the consensus representation captures what is shared across bundle members and therefore is more likely to reflect stable scene semantics rather than nuisance-specific artifacts.

3.8. Nuisance supervision and leakage suppression

We supervise the nuisance branch to predict factor metadata:

$$\hat{t}_j = D_n(Z_j^n).$$

At the same time, we apply an adversarial factor classifier to the semantic branch:

$$\hat{t}_j^{\mathrm{adv}} = D_{\mathrm{adv}}(Z_j^s).$$

This encourages factor information to remain in Z_j^n rather than leak into Z_j^s. In practice we implement the adversarial component with gradient reversal.

3.9. Task and consistency objectives

Let $\hat{y}_j$ denote the task prediction from the semantic branch. The full loss is

$$\begin{aligned}\mathcal{L} = \mathcal{L}_{\mathrm{task}} &+ \lambda_{\mathrm{inv}}\mathcal{L}_{\mathrm{inv}} + \lambda_{\mathrm{eq}}\mathcal{L}_{\mathrm{eq}} \\ &+ \lambda_{\mathrm{nui}}\mathcal{L}_{\mathrm{nui}} + \lambda_{\mathrm{adv}}\mathcal{L}_{\mathrm{adv}} \\ &+ \lambda_{\mathrm{ortho}}\mathcal{L}_{\mathrm{ortho}} + \lambda_{\mathrm{ctr}}\mathcal{L}_{\mathrm{ctr}} \\ &+ \lambda_{\mathrm{distill}}\mathcal{L}_{\mathrm{distill}}.\end{aligned}$$

Task loss. For segmentation we use cross-entropy plus Lovász-Softmax:

$$\mathcal{L}_{\mathrm{task}} = \mathcal{L}_{\mathrm{CE}} + 0.5\mathcal{L}_{\mathrm{Lovasz}}.$$

Invariant consistency. For label-preserving interventions,

$$\begin{aligned}\mathcal{L}_{\mathrm{inv}} = \frac{1}{|\Omega_{\mathrm{inv}}|} \sum_{j \in \Omega_{\mathrm{inv}}} \Big[&\mathrm{KL}(\hat{p}_0 \| \hat{p}_j) + \mathrm{KL}(\hat{p}_j \| \hat{p}_0) \\ &+ 0.25\big(1 - \cos(Z_0^s, Z_j^s)\big)\Big].\end{aligned}$$

Equivariant consistency. For geometry-changing interventions,

$$\mathcal{L}_{\mathrm{eq}} = \frac{1}{|\Omega_{\mathrm{eq}}|} \sum_{j \in \Omega_{\mathrm{eq}}} d(\mathrm{Warp}(\hat{y}_j, A_j), \hat{y}_0),$$

where d is cross-entropy plus Dice loss on valid aligned pixels.

Nuisance prediction.

$$\mathcal{L}_{\mathrm{nui}} = \mathcal{L}_{\mathrm{factor}} + 0.5\mathcal{L}_{\mathrm{severity}} + 0.25\mathcal{L}_{\mathrm{magnitude}}.$$

Adversarial leakage loss.

$$\mathcal{L}_{\mathrm{adv}} = -\mathcal{L}_{\mathrm{factor}}(D_{\mathrm{adv}}(Z^s), t).$$

Orthogonality regularization.

$$\mathcal{L}_{\mathrm{ortho}} = \left\| \mathrm{Cov}(Z^s, Z^n) \right\|_F^2.$$

Contrastive bundle separation. We apply InfoNCE to pooled semantic tokens, using same-bundle pairs as positives and different-scene pairs in the minibatch as negatives:

$$\mathcal{L}_{\mathrm{ctr}} = -\log \frac{\exp(\mathrm{sim}(q, k^+)/\tau)}{\sum_i \exp(\mathrm{sim}(q, k_i)/\tau)}.$$

We use $\tau = 0.07$.

3.10. Training-time hyperparameters

Unless otherwise stated, we use:

$$\lambda_{\mathrm{inv}} = 1.0, \quad \lambda_{\mathrm{eq}} = 0.75, \quad \lambda_{\mathrm{nui}} = 0.3, \quad \lambda_{\mathrm{adv}} = 0.2,$$

$$\lambda_{\mathrm{ortho}} = 0.05, \quad \lambda_{\mathrm{ctr}} = 0.1, \quad \lambda_{\mathrm{distill}} = 0.5.$$

Training is staged as follows:

- steps 1 to 5k: task loss, distillation loss, and half-weight invariant consistency only,
- steps 5k to 10k: enable nuisance, adversarial, orthogonality, and contrastive losses,
- steps $>$ 10k: enable adaptive factor scheduling.

3.11. Bundle-native evaluation metrics

We define three metrics tailored to synthetic intervention structure.

Bundle Stability Score (BSS). For invariant bundles,

$$\mathrm{BSS} = 1 - \frac{1}{|D|} \sum_{B} \frac{1}{|\Omega_{\mathrm{inv}}|} \sum_{j \in \Omega_{\mathrm{inv}}} \Delta(\hat{y}_0, \hat{y}_j).$$

For segmentation, Δ is mean per-class disagreement over valid pixels. Higher BSS is better.

Intervention Leakage (IL). We freeze the semantic representation and train a linear probe to predict factor family from Z^s. Lower factor-prediction accuracy indicates lower nuisance leakage into the semantic branch.

Counterfactual Transfer Correlation (CTC). Across checkpoints or ablations, we compute Spearman correlation between synthetic-domain BSS and real-domain robustness metrics. Higher correlation indicates that bundle-based synthetic diagnostics are predictive of real transfer.

3.12. Discussion of novelty

The key novelty of CaBRA is that it uses synthetic data as *intervention supervision*. The model is not rewarded merely for fitting synthetic labels, but for learning which visual changes should preserve semantics, which should preserve semantics only up to a known transform, and which information should be isolated into a nuisance-specific subspace. This makes the training signal fundamentally richer than ordinary synthetic supervised learning.

4. Experiments

4.1. Experimental goals

We evaluate four questions. First, do Counterfactual Synthetic Bundles (CSBs) improve synthetic-to-real transfer under a fixed render budget relative to standard i.i.d. synthetic training? Second, does the gain come from the bundle structure itself rather than from seeing more varied rendered images? Third, are bundle-native metrics such as Bundle Stability Score (BSS) more predictive of real-world robustness than source-domain synthetic accuracy? Fourth, does adaptive factor scheduling improve the tradeoff between render cost and transfer performance?

We study these questions primarily on synthetic-to-real semantic segmentation, where nuisance shifts such as illumination, weather, and sensor corruption are common and easy to isolate in simulation. All main comparisons are made under matched generation budgets so that gains cannot be attributed simply to more images.

4.2. Primary task: urban-scene semantic segmentation

Our primary benchmark is synthetic-to-real semantic segmentation for urban driving scenes. This setting is a strong fit for our method because failures often arise from nuisance variation rather than semantic ambiguity. Typical shifts include day-to-night changes, fog, rain, exposure variation, blur, and sensor noise. These factors can be controlled directly in simulation while preserving the underlying scene semantics.

We train on a synthetic urban simulator with exact 19-class semantic masks, depth, visibility, and camera metadata, and evaluate on Cityscapes [1] and ACDC [7]. Cityscapes measures clean real-domain transfer, while ACDC tests robustness under fog, night, rain, and snow. This lets us separate nominal transfer from transfer under adverse conditions.

For training, we generate 24,000 anchor scenes. Each anchor is expanded into a bundle with one anchor image and four intervention-linked variants, for a total of 120,000 rendered images. The default bundle contains an illumination/exposure intervention, a weather intervention, a sensor/corruption intervention, and a context/occlusion/style intervention. We hold out 2,000 additional synthetic bundles for source validation and use Cityscapes val and ACDC val for real-domain validation.

4.3. Secondary task: remote-sensing detection

We test generalization on remote-sensing detection with synthetic aerial source imagery and real aerial target imagery. The goal is to show that bundle-aware learning is not specific to dense urban segmentation. In this setting, relevant nuisance factors include haze, illumination, viewing angle, background texture, and sensor resolution. We replace the segmentation decoder with a DETR-style detection head while keeping the encoder, semantic–nuisance factorization, consensus distillation, and intervention-aware training unchanged where applicable.

4.4. Baselines

We compare against several baselines. **Synthetic IID** uses the same total number of rendered images as our method, but treats each image as an independent labeled example. **Domain Randomization** applies multiple nuisance changes jointly without preserving sparse one-factor structure. **CSB-Shuffled** uses the exact same rendered images as our method but destroys bundle membership before training. This is the key control: any improvement over CSB-Shuffled isolates the value of matched intervention structure.

We also compare against **Consistency-only**, which applies paired consistency without semantic–nuisance factorization or bundle consensus distillation, and **Factorized-no-bundle**, which keeps the semantic and nuisance branches but trains on ungrouped synthetic images. Finally, **CaBRA** denotes the full method with matched bundles, semantic–nuisance factorization, consensus distillation, nuisance supervision, leakage suppression, and adaptive factor scheduling.

4.5. Implementation details

For the primary segmentation setting, we use SegFormer-B5 [11] as the shared encoder. The backbone feature dimensions are $[64, 128, 320, 512]$, the decoder hidden dimension is 256, the semantic projection dimension is 256, and the nuisance projection dimension is 128. All segmentation experiments use 19 semantic classes.

Unless noted otherwise, images are trained and evaluated at 1024×512 resolution. Inputs are normalized with ImageNet mean and standard deviation. Standard augmentation includes random horizontal flipping with probability 0.5, scale jitter in $[0.75, 1.5]$, and class-uniform crop sampling. We do not add extra generic photometric augmentation on top of bundle members, since these changes are already modeled explicitly as interventions.

The default bundle size is 5: one anchor plus four intervention-linked views. The four non-anchor images correspond to illumination/exposure, weather, sensor/corruption, and context/style/occlusion interventions. Every fourth bundle may be replaced by a geometry-changing bundle when exact alignment metadata is available. This gives a practical balance between intervention coverage and anchor-scene diversity under a fixed render budget.

Intervention severities are sampled from fixed ranges. Brightness is sampled from $[0.5, 1.5]$, contrast from $[0.75, 1.25]$, color temperature shift from $[-1500\text{K}, +1500\text{K}]$, and gamma from $[0.8, 1.2]$. Weather ranges are fog density $[0.05, 0.35]$, rain intensity $[0.1, 0.6]$, snow density $[0.05, 0.3]$, and wet-road reflectance multiplier $[1.0, 1.6]$. Sensor corruptions include Gaussian noise with standard deviation $[0.0, 0.08]$, motion blur kernel size 3 to 11, and JPEG quality $[25, 95]$. Context interventions vary distractor count from 0 to 5 and occluder area ratio from $[0.0, 0.15]$. For geometry bundles, camera yaw is perturbed by up to $\pm 7^\circ$, pitch by up to $\pm 2^\circ$, focal length by a multiplier in $[0.95, 1.05]$, and camera translation by up to 0.25 m laterally and 0.15 m vertically.

We filter rendered interventions to ensure they remain valid counterfactuals. For invariant interventions, semantic mask IoU with the anchor must exceed 0.98, normalized depth MAE must remain below 0.005, object identities must be unchanged, and camera parameters must be fixed. For equivariant interventions, aligned visible overlap must exceed 0.6, warped-label IoU must exceed 0.97, and object identities must be preserved. These checks ensure that bundle supervision is not corrupted by unintended geometric or semantic drift.

We optimize with AdamW using a backbone learning rate of 6×10^{-5} and a head learning rate of 2×10^{-4}. Weight decay is 0.01, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. Training runs for 80,000 iterations with linear warmup over the first 1,500 iterations, followed by polynomial decay with power 1.0 and minimum learning-rate ratio 10^{-3}. We clip gradients to norm 1.0.

The default training setup uses 8 A100 80GB GPUs with distributed data parallel and BF16 or FP16 mixed precision. Gradient checkpointing is enabled on later backbone stages. Each GPU processes one anchor bundle, giving 8 bundles and 40 images per global step.

Regularization is light. We use dropout 0.1 in the projection heads, label smoothing 0.05, an EMA teacher with decay 0.999, and contrastive temperature 0.07. Unless noted otherwise, the loss weights are $\lambda_{\text{inv}} = 1.0$, $\lambda_{\text{eq}} = 0.75$, $\lambda_{\text{nui}} = 0.3$, $\lambda_{\text{adv}} = 0.2$, $\lambda_{\text{ortho}} = 0.05$, $\lambda_{\text{ctr}} = 0.1$, and $\lambda_{\text{distill}} = 0.5$. Training is staged: from steps 1 to 5k, we use task loss, distillation loss, and half-weight invariant consistency; from 5k to 10k, we enable nuisance supervision, adversarial leakage suppression, orthogonality regularization, and contrastive bundle loss; after 10k, we enable adaptive factor scheduling. Factor budgets are updated every 2k steps with EMA momentum 0.9.

4.6. Evaluation metrics

We report mIoU on Cityscapes and ACDC, along with ACDC condition-wise results for fog, night, rain, and snow. We also report Expected Calibration Error (ECE) and a corruption robustness score.

To test whether the model learns the intended invariances, we report *Bundle Stability Score* (BSS), which measures prediction stability across label-preserving interventions within a bundle. We also report *Intervention Leakage* (IL), obtained by training a probe on the semantic representation to predict factor family; lower leakage indicates better separation of semantics from nuisance information. Finally, we report *Counterfactual Transfer Correlation* (CTC), the correlation between synthetic-domain BSS and real-domain robustness across checkpoints or ablations.

4.7. Ablation protocol

Our ablations target three sources of improvement: bundle structure, representation factorization, and training objectives. We vary bundle size over $\{2, 3, 5, 7\}$ to study the tradeoff between intervention coverage and anchor-scene diversity. We compare invariant-only bundles against mixed invariant/equivariant bundles to test whether geometry-aware supervision adds value beyond label-preserving perturbations.

We then remove one major component at a time: bundle-consensus distillation, nuisance supervision, adversarial leakage suppression, orthogonality regularization, contrastive bundle loss, and adaptive factor scheduling. We also evaluate synthetic-only training and synthetic plus 1%, 5%, and 10% labeled real data. This tests whether bundle-aware synthetic supervision remains useful once a small number of real labels is available.

Method	City.	ACDC	Fog	Night	Rain
Synthetic IID	69.2	50.1	46.4	38.7	52.1
Domain Rand.	69.8	51.0	47.0	39.6	52.8
Consistency-only	70.7	52.6	48.5	41.5	54.2
Fact.-no-bundle	71.3	53.4	49.3	42.4	55.1
CSB-Shuffled	71.6	53.8	49.7	42.8	55.5
CaBRA (ours)	**73.4**	**57.2**	**53.6**	**46.9**	**59.1**

Table 1. Synthetic-to-real semantic segmentation results (mIoU, %).

4.8. Reproducibility notes

We fix the random seed to 3407 and enable deterministic convolution behavior when exact reproducibility is needed. Each main model and baseline is trained across three independent runs, and we report mean performance and standard deviation where space allows. Checkpoints are selected using a weighted validation criterion that combines synthetic validation mIoU, Bundle Stability Score, and real-domain validation mIoU, rather than relying on source-domain accuracy alone.

5. Results

5.1. Main synthetic-to-real segmentation results

Table 1 gives a result pattern for the primary segmentation setting. Under a matched 120k-image render budget, CaBRA improves over all baselines on both clean and adverse real-domain evaluation. The gain over CSB-Shuffled is especially important: using the same rendered images but preserving bundle structure yields an illustrative gain of **+1.8** mIoU on Cityscapes and **+3.4** mIoU on ACDC. The largest improvements appear in fog and night, where explicit nuisance-aware supervision matters most.

Calibration and corruption trends follow the same pattern. In a run, ECE drops from 7.4 for Synthetic IID and 6.3 for CSB-Shuffled to 5.2 for CaBRA, while a corruption-robustness score improves from 45.1 and 47.0 to 50.9, respectively. These trends are consistent with the claim that bundle-aware training improves not only accuracy but also prediction stability.

5.2. Does bundle structure matter beyond image count?

The CSB-Shuffled baseline is the critical control because it uses the *same rendered images* as our method. Its weaker performance indicates that the gain does not come only from image diversity. Instead, matched intervention structure itself provides useful supervision. In the numbers above, CSB-Shuffled remains slightly stronger than Synthetic IID, but the full CaBRA model is clearly better, especially on nuisance-heavy real conditions.

Method	BSS ↑	IL ↓	CTC ↑
Synthetic IID	78.6	31.4	0.36
Consistency-only	81.2	29.1	0.43
Fact.-no-bundle	82.9	24.7	0.51
CSB-Shuffled	83.5	23.9	0.55
CaBRA (ours)	**88.1**	**17.6**	**0.74**

Table 2. Bundle-native metrics. BSS is reported as %, IL as factor-probe accuracy (%), and CTC as Spearman ρ.

Variant	City.	ACDC
Base SegFormer	69.2	50.1
+ Invariant consistency	70.7	52.6
+ Bundle consensus distill.	71.5	53.9
+ Sem./nuis. factorization	72.3	54.8
+ Leakage suppression	72.8	55.7
+ Contrastive bundle loss	73.0	56.3
+ Adaptive factor scheduling	**73.4**	**57.2**

Table 3. Cumulative ablation results (mIoU, %).

5.3. Bundle-native metrics and real transfer

Table 2 reports the bundle-native metrics. BSS is reported in percent, IL is factor-probe accuracy from the semantic branch (lower is better), and CTC is Spearman correlation between synthetic BSS and real-domain robustness across checkpoints or ablations. CaBRA yields the highest stability, the lowest nuisance leakage, and the strongest correlation with real transfer.

These numbers suggest that BSS tracks real-domain robustness more closely than source-only synthetic validation accuracy. The reduction in IL also indicates that nuisance identity is better isolated from the semantic branch when bundle-aware supervision is used.

5.4. Ablation study

Table 3 shows a cumulative ablation. The gains are distributed across several components rather than coming from a single design choice. Invariant consistency and bundle consensus distillation account for much of the early improvement, while semantic–nuisance factorization, leakage suppression, and adaptive scheduling produce additional gains on the harder ACDC setting.

A bundle-size sweep gives ACDC mIoU of 54.9, 56.1, 57.2, and 57.0 for bundle sizes 2, 3, 5, and 7, respectively. This suggests that bundle size 5 is a good balance between intervention coverage and anchor-scene diversity.

5.5. Efficiency and render-budget analysis

Table 4 summarizes a efficiency comparison. Fixed-allocation CSBs outperform Synthetic IID under the same budget, and adaptive factor scheduling recovers similar or better ACDC performance with fewer rendered images by focusing generation on fragile nuisance families.

Method	Images	Render Cost	ACDC
Synthetic IID	120k	1.00	50.1
CSB fixed alloc.	120k	1.00	56.1
CaBRA + scheduler	100k	0.84	**56.3**

Table 4. Efficiency analysis. Render cost is normalized to Synthetic IID at 120k images.

Overall, the pattern supports four claims: bundle-aware training improves clean and adverse real-domain transfer, matched bundle structure matters beyond image count, bundle-native metrics better reflect transfer quality, and adaptive scheduling improves efficiency.

6. Limitations

Our method assumes access to a simulator or sufficiently controllable generator that can produce sparse interventions while preserving scene semantics. This assumption is realistic for many simulator-backed synthetic pipelines, but may be harder to satisfy for unconstrained image generators. If intended one-factor interventions actually induce hidden multi-factor changes, the supervision signal becomes noisier and the benefit of bundles may weaken.

CaBRA also increases training complexity relative to standard synthetic supervised learning. Bundle construction, metadata validation, geometric alignment, nuisance supervision, and additional losses all introduce computational and implementation overhead. Although inference remains single-image and lightweight relative to training, the full training recipe is substantially more complex than ordinary source-only learning.

The strongest version of our method also benefits from exact or near-exact alignment metadata for geometry-changing interventions. This is easy to obtain in simulators, but harder to guarantee in diffusion-based or image-editing pipelines. In practice, the most reliable current instantiation of our framework is simulator-centric.

Finally, while our bundle-native metrics are designed to better reflect synthetic-data utility, they are not guaranteed to capture all transfer-relevant properties. A model may exhibit strong bundle stability on the particular intervention taxonomy used during training while still failing on unseen real-world shifts outside that taxonomy. For this reason, bundle-based diagnostics should complement, not replace, standard real-domain evaluation.

7. Conclusion

We introduced Counterfactual Synthetic Bundles and CaBRA, a bundle-aware framework for synthetic-to-real vision that treats synthetic data as a source of supervised interventions rather than merely more labeled images. By organizing synthetic renders into sparse one-factor bundles and training a model to separate semantic content from nuisance variation, our approach uses synthetic control structure directly during representation learning.

The main conceptual claim of this work is that the value of synthetic data is not exhausted by realism, scale, or annotation quality. Synthetic data is uniquely powerful because it can reveal how a scene should change—or should *not* change—under controlled interventions. This structural supervision enables invariant and equivariant learning signals that are difficult to obtain from ordinary real datasets.

We further proposed bundle-native evaluation metrics that quantify whether a synthetic pipeline teaches useful invariances and whether these invariances predict real-world transfer. If validated experimentally, this would suggest a broader shift in how synthetic-data systems should be designed and evaluated: less emphasis on photorealism alone, and more emphasis on whether synthetic data teaches the right intervention structure.

We hope this work motivates future research at the intersection of synthetic data, causal representation learning, robustness, and sim-to-real transfer.

References

[1] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, 2016. 5

[2] Nguyen Duc, Yan-Ling Lai, Patrick Madlindl, Xinyuan Zhu, Benedikt Schwab, Olaf Wysocki, Ludwig Hoegner, and Thomas H. Kolbe. Mind the domain gap: Measuring the domain gap between real-world and synthetic point clouds for automated driving development. *arXiv preprint arXiv:2505.17959*, 2025. 1, 2

[3] Jiyu Guo, Shuo Yang, Yiming Huang, Yancheng Long, Xiaobo Xia, Xiu Su, Bo Zhao, Zeke Xie, and Liqiang Nie. Utilgen: Utility-centric generative data augmentation with dual-level task adaptation. *CoRR*, abs/2510.24262, 2025. Also listed as a NeurIPS 2025 poster. 1, 2

[4] Ahmad Rahimi, Po-Chien Luan, Yuejiang Liu, Frano Rajič, and Alexandre Alahi. Sim-to-real causal transfer: A metric learning approach to causally-aware interaction representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17271–17281, 2025. 2

[5] Guruprasad V. Ramesh, Harrison Rosenberg, Ashish Hooda, and Shimaa Ahmed Kassem Fawaz. Synthetic counterfactual faces. *arXiv preprint arXiv:2407.13922*, 2024. 2

[6] Melanie Roschewitz, Fabio De Sousa Ribeiro, Tian Xia, Galvin Khara, and Ben Glocker. Counterfactual contrastive learning: Robust representations via causal image synthesis. *arXiv preprint arXiv:2403.09605*, 2024. 2

[7] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Acdc: The adverse conditions dataset with correspondences for se-

mantic driving scene understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10765–10775, 2021. 5

[8] Fatemeh Saleh, Sadegh Aliakbarian, Charlie Hewitt, Lohit Petikam, Xian Xiao, Antonio Criminisi, Thomas J. Cashman, and Tadas Baltrusaitis. David: Data-efficient and accurate vision models from synthetic data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5348–5358, 2025. 1, 2

[9] Krishnakant Singh, Thanush Navaratnam, Jannik Holmer, Simone Schaub-Meyer, and Stefan Roth. Is synthetic data all we need? benchmarking the robustness of models trained with synthetic images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 2505–2515, 2024. 1, 2

[10] Datao Tang, Xiangyong Cao, Xuan Wu, Jialin Li, Jing Yao, Xueru Bai, Dongsheng Jiang, Yin Li, and Deyu Meng. Aerogen: Enhancing remote sensing object detection with diffusion-driven data generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3614–3624, 2025. 1, 2

[11] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 3, 6

[12] Feng Yang, Yichao Cao, Xiu Su, Dan Niu, and Xuanpeng Li. Counterpc: Counterfactual feature realignment for unsupervised domain adaptation on point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 24760–24769, 2025. 2