Target-Aware Nuisance Shaping for Object Detection

Tianyi Xu[1] **Yi Niu**[1] **Xinkun Wang**[1] **Yanbiao Ma**[2]
Mingming Ma[1*] **QingYu Luo**[1] **Fu Zou**[1] **Fu Li**[1]
[1]Xidian University
[2]Gaoling School of Artificial Intelligence, Renmin University of China
{25171111417,25171213993,23171110716,23171110141}@stu.xidian.edu.cn
{niuyi,fuli}@mail.xidian.edu.cn
mamingming@xidian.edu.cn, ybma1998@ruc.edu.cn

Abstract

Object detectors are expected to recognize objects independently of intrinsic object-level attributes, such as scale and local visibility. Yet, natural detection datasets contain systematic intrinsic attribute biases, and modern detectors inevitably entangle these biases within their learned representation geometry, leading to non-uniform performance across the continuous attribute spectrum. This observation exposes a central limitation of conventional data augmentation: while attribute-blind transformations enlarge the training distribution, they fail to explicitly correct the dataset-inherent nuisance structure. To address this, we propose **Target-Aware Nuisance Shaping** (TANS), a unified framework that re-purposes data augmentation as an object-conditioned vicinal distribution design. TANS processes training samples through a serial pipeline of two fundamental operators: first, **Scale Prior Shaping** (SPS) geometrically transports object scales toward low-density regions of the natural scale distribution via class-conditional log-area priors; subsequently, **Local Visibility Shaping** (LVS) photometrically modulates local foreground-background contrast by constructing smooth, box-conditioned Gaussian fields, strictly preserving the newly established geometry and labels. Furthermore, a feature-state controller monitors the evolving representation geometry to adaptively adjust the shaping strength during training. Extensive experiments on MS COCO and Pascal VOC demonstrate that TANS significantly improves detection performance over strong augmentation baselines across modern architectures. Specifically, on COCO with YOLOv11, TANS achieves a 3.6 mAP gain over the Mosaic baseline. Beyond aggregate accuracy, our attribute-regime analyses, feature-state dynamics, schedule comparisons, and logit-space probes consistently indicate that TANS effectively decouples the detector from intrinsic dataset biases, establishing target-aware shaping as a rigorous paradigm for learning attribute-invariant representations.

1 Introduction

Modern object detection pipelines [1–6] rely on discriminative representations learned from large annotated datasets [7–10]. A robust detector should recognize an object independently of intrinsic object-level attributes, such as geometric scale and local visibility. Natural detection datasets, however, are not attribute-neutral [11–13]. Categories often occupy biased regions of the intrinsic attribute spectrum [14, 15], and detectors can encode these dataset-inherent structures into their representation space [16, 17].

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

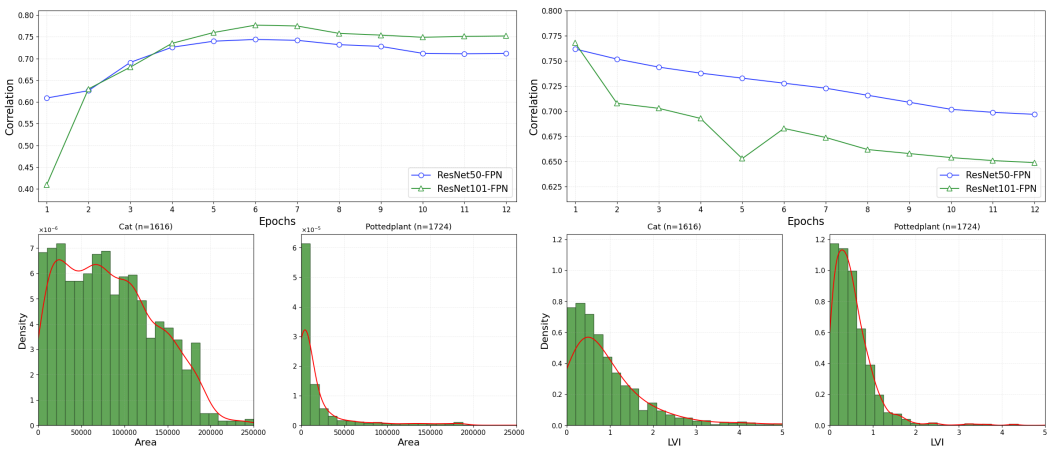


Figure 1: Representation entanglement with intrinsic attributes. Category AP remains strongly correlated with scale/area and Local Visibility Index (LVI) during training on Pascal VOC. The density plots compare high-performing and low-performing categories, showing that categories occupy different regions of the intrinsic attribute spectrum.

We study this phenomenon through two universal attributes in detection: scale, measured by bounding-box area [9, 11], and local visibility, measured by the proposed Local Visibility Index (LVI). LVI is an instance-level visibility measurement rather than a dataset prior. It estimates the local foreground-background separability of each annotated object and provides a visibility signal for nuisance shaping, representation diagnostics, and controller feedback. As shown in Fig. 1, category AP remains strongly correlated with scale and LVI during training. This persistent correlation suggests that the detector partially relies on intrinsic dataset structure as a shortcut, instead of learning fully attribute-invariant object representations.

This observation motivates a different view of data augmentation. Standard compositional augmentations, including MixUp [18], CutMix [19], Mosaic, and Simple Copy-Paste [20], are effective at expanding the vicinal distribution. Yet they are largely attribute-blind. They apply generic transformations without explicitly asking whether the generated samples reduce the coupling between detector behavior and dataset-inherent attributes.

We address this limitation with Target-Aware Nuisance Shaping (TANS), an object-conditioned vicinal distribution design for object detection. TANS uses two complementary operators. Scale Prior Shaping (SPS) estimates class-conditional log-area priors and transports object scales toward undercovered regions of the empirical scale distribution. Local Visibility Shaping (LVS) measures local foreground-background visibility through LVI and constructs smooth box-conditioned Gaussian fields to modulate local contrast while preserving labels and geometry. A feature-state controller further monitors representation geometry, including foreground-background margin, effective rank, and feature volume, and adapts the shaping strength during training.

Our contributions are summarized as follows:

- We reveal intrinsic attribute bias as a source of representation entanglement in object detection, using scale and local visibility as diagnostic coordinates of the attribute spectrum.
- We propose TANS, a training-time framework that turns augmentation into object-conditioned nuisance shaping through SPS for geometric scale transport and LVS for local visibility shaping.
- We introduce a feature-state controller and validate TANS across modern detection settings. Benchmark results, attribute-regime diagnostics, schedule analyses, and logit-space probes show that TANS reduces attribute-conditioned ambiguity beyond standard attribute-blind augmentation.

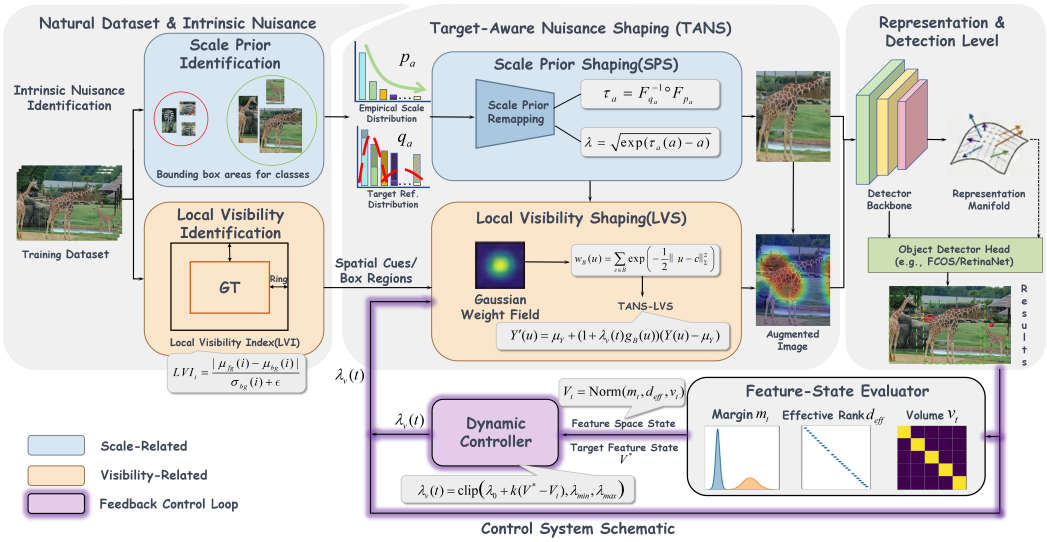


Figure 2: Target-Aware Nuisance Shaping (TANS). TANS re-purposes data augmentation as object-conditioned vicinal distribution design. The SPS branch estimates class-conditional scale priors and transports object scales along the log-area attribute axis. The LVS branch measures local visibility through an instance-level LVI signal and constructs smooth box-conditioned Gaussian fields to shape local foreground-background contrast on the updated geometry. A feature-state evaluator monitors representation geometry and drives the dynamic controller for adaptive shaping strength.

2 Related Work

Generic data augmentation as vicinal design. Data augmentation [21–26] is a fundamental tool for improving the generalization of deep neural networks. In object detection, region-based vicinal designs such as Mixup [18], Cutout [27], CutMix [19], Mosaic, and Simple Copy-Paste [20] have become standard training components. These methods enlarge the neighborhood of training samples, but most of them remain attribute-blind: the transformation distribution is not explicitly tied to the intrinsic attribute structure of the dataset or to the representation geometry learned by the detector.

Object-aware and attribute-conditioned augmentation. Recent works introduce stronger object awareness into augmentation. Scale-Aware AutoAug [28] searches for scale-specific policies, controllable diffusion augmentation [29] synthesizes object-level variations, ReCon [30] performs region-controllable augmentation, and OA-DG [31] uses object-centric mixing to improve domain generalization. These approaches demonstrate the value of attribute-aware transformations, but often require policy search, generative priors, or domain-specific assumptions. TANS instead treats augmentation as a lightweight object-conditioned nuisance-shaping problem guided by measured attribute states, empirical scale priors, and feature-state feedback.

GMM-based equalization improves image contrast [32], while color/illuminance augmentation and exposure synthesis support re-identification [33] and classification [34]. TANS instead shapes box-conditioned contrast during detector training, jointly with a class-conditional scale prior and feature-state feedback.

Intrinsic nuisance factors and representation bias. Scale imbalance is a long-recognized issue in detection [11–13, 35–37], while visibility-related problems are often studied in specialized settings such as low-light or adverse weather detection. Our work views scale and local visibility as universal intrinsic attributes present in natural detection datasets. Rather than treating them as isolated engineering difficulties, we study how they become entangled with the detector's learned representation and use target-aware shaping to reduce this dependence.

3 Target-Aware Nuisance Shaping

3.1 Unified Formulation: Object-Conditioned Vicinal Risk

Most detection augmentations [18–21] sample transformations from a label-agnostic distribution. Although such transformations enlarge the training neighborhood, they do not explicitly control how the generated samples cover dataset-inherent attribute structure. We formulate augmentation as *Target-Aware Nuisance Shaping* (TANS), where the transformation distribution is conditioned on annotated objects and on the evolving representation state of the detector.

Let $\mathcal{D} = \{(x_n, \mathcal{B}_n, \mathcal{Y}_n)\}_{n=1}^{N}$ be a detection dataset, where x_n is an image, $\mathcal{B}_n = \{b_{n,i}\}_{i=1}^{M_n}$ denotes ground-truth boxes, and $\mathcal{Y}_n = \{y_{n,i}\}_{i=1}^{M_n}$ contains category labels. For an object box $b_i = (x_i^1, y_i^1, x_i^2, y_i^2)$ in an image of height H and width W, we define its log-relative area as

$$a_i = \log\left(\frac{(x_i^2 - x_i^1)(y_i^2 - y_i^1)}{HW} + \epsilon\right), \tag{1}$$

where ϵ is a small constant for numerical stability. Beyond geometric scale, we use local visibility to describe how strongly an object is separated from its local background.

Let T_θ be a detection-preserving augmentation operator. Instead of sampling θ from a fixed random distribution, TANS samples from an object-conditioned distribution $\theta_t \sim \pi_t(\theta \mid x, \mathcal{B}, \mathcal{Y}, C_t)$, where C_t is the feature-state signal at training step t. The resulting vicinal objective is

$$\widetilde{\mathcal{R}}(\phi) = \mathbb{E}_{(x,\mathcal{B},\mathcal{Y})\sim\mathcal{D}}\mathbb{E}_{\theta_t\sim\pi_t}\left[\mathcal{L}\big(f_\phi(T_{\theta_t}(x,\mathcal{B})),\mathcal{Y}\big)\right]. \tag{2}$$

As illustrated in Fig. 2, TANS instantiates this objective through a serial SPS→LVS pipeline. This order is design-motivated rather than claimed as a mathematical necessity: SPS first determines the final object geometry and local spatial support, and LVS then constructs the visibility field on this updated geometry. Applying LVS before geometric transport would require the photometric field to be warped with the object crop, which can introduce field misalignment near object boundaries. We therefore use SPS→LVS as the default implementation in all experiments.

3.2 Scale Prior Shaping (SPS) via Monotone Transport

Scale variation in object detection is continuous, category-dependent, and highly non-uniform. Uniform random resizing cannot distinguish dominant scale ranges from low-density regions of the empirical scale distribution. SPS addresses this issue by remapping the class-conditional log-area prior toward a reference distribution that allocates more probability mass to under-covered scale states.

For category c, let $\mathcal{A}_c = \{a_i \mid y_i = c\}$ be the set of log-relative object areas. We estimate the empirical log-scale density $p_a^{(c)}(a)$ using a one-dimensional Gaussian mixture model [38]:

$$p_a^{(c)}(a) = \sum_{k=1}^{K_c} \pi_{c,k}\mathcal{N}(a \mid \mu_{c,k}, \sigma_{c,k}^2). \tag{3}$$

The number of mixture components is selected by the Bayesian Information Criterion:

$$\mathrm{BIC}_c(K) = -2\log\mathcal{L}_c(K) + (3K-1)\log|\mathcal{A}_c|, \tag{4}$$

where $\mathcal{L}_c(K)$ is the maximized likelihood. To avoid amplifying noisy extremes, we define a clipped support $\mathcal{I}_c = [Q_\eta^{(c)}, Q_{1-\eta}^{(c)}]$ and construct a smoothed inverse-density reference prior:

$$q_a^{(c)}(a) = \frac{\left[p_{\max}^{(c)} - p_a^{(c)}(a)\right]_+ + \epsilon_q}{Z_c}, \qquad a \in \mathcal{I}_c. \tag{5}$$

Let $F_p^{(c)}$ and $F_q^{(c)}$ be the cumulative distribution functions of $p_a^{(c)}$ and $q_a^{(c)}$. SPS defines the monotone transport map

$$\tau_a^{(c)}(a) = \left(F_q^{(c)}\right)^{-1}\left(F_p^{(c)}(a)\right). \tag{6}$$

The target log-area is $a_i^\star = \tau_a^{(c)}(a_i)$, and the isotropic scale factor is

$$s_i = \sqrt{\exp(a_i^\star - a_i)}. \tag{7}$$

We clip s_i to $[s_{\min}, s_{\max}]$ and reject pasted candidates that are too small, exceed the valid image region after clipping, or overlap existing objects above the IoU threshold. These geometric-validity checks maintain consistent images and annotations after composition; target awareness enters through the class-conditional scale-proposal distribution.

3.3 Local Visibility Shaping (LVS) via Smooth Perturbation Fields

Scale is not the only intrinsic attribute that affects detection. An object can have a favorable size while remaining photometrically entangled with its local background. We therefore introduce Local Visibility Shaping (LVS), which uses an instance-level Local Visibility Index (LVI) as a measurement signal for local foreground-background separability. Unlike SPS, LVI is not a dataset-level prior. It is computed from the current annotated image and used to characterize the visibility state that LVS and the feature-state controller act upon.

Let $Y_x(u) \in [0,1]$ be the luminance channel. For an object b_i with center (c_i^x, c_i^y) and dimensions (w_i, h_i), we define a Gaussian foreground weight inside the box:

$$G_i^{\mathrm{fg}}(u) = \exp\left[-\frac{1}{2}\left(\frac{(u_x - c_i^x)^2}{\sigma_{x,i}^2} + \frac{(u_y - c_i^y)^2}{\sigma_{y,i}^2}\right)\right], \qquad \sigma_{x,i} = 0.25 w_i, \quad \sigma_{y,i} = 0.25 h_i. \tag{8}$$

The weighted foreground luminance is

$$\mu_{\mathrm{fg}}(i) = \frac{\sum_{u \in b_i} G_i^{\mathrm{fg}}(u) Y_x(u)}{\sum_{u \in b_i} G_i^{\mathrm{fg}}(u) + \epsilon}. \tag{9}$$

For the local background, we construct an outer box with the same center and side lengths $\sqrt{2}w_i$ and $\sqrt{2}h_i$. The ring is the outer box minus the foreground box, excluding pixels covered by other annotated objects. This makes the ring area equal to the foreground box area. Denoting this ring by R_i, we compute

$$\mu_{\mathrm{bg}}(i) = \frac{1}{|R_i|}\sum_{u \in R_i} Y_x(u), \qquad \sigma_{\mathrm{bg}}(i) = \sqrt{\frac{1}{|R_i|}\sum_{u \in R_i}\big(Y_x(u) - \mu_{\mathrm{bg}}(i)\big)^2} + \epsilon. \tag{10}$$

The Local Visibility Index is

$$\mathrm{LVI}_i = \frac{|\mu_{\mathrm{fg}}(i) - \mu_{\mathrm{bg}}(i)|}{\sigma_{\mathrm{bg}}(i) + \epsilon}. \tag{11}$$

A smaller LVI indicates weaker local foreground-background separation. LVI measures this attribute and defines the controller's low-LVI diagnostic set; LVS perturbs it through the box-conditioned field.

LVS constructs a continuous spatial perturbation field from the annotated boxes. For each object, let $G_i(u)$ be a Gaussian field centered at the box center and aligned with the object size. In the main implementation, all annotated objects contribute uniformly to the field, and LVI is not used as a hand-crafted per-instance priority weight. The fused field is

$$w_{\mathcal{B}}(u) = 1 - \prod_{i=1}^{M}\big(1 - G_i(u)\big). \tag{12}$$

The field is converted into a zero-mean normalized field $g_{\mathcal{B}}(u) \in [-1, 1]$ by subtracting the spatial mean and dividing by the maximum absolute deviation. LVS then applies a luminance-only local contrast transformation:

$$Y_x'(u) = \mathrm{clip}\left[\mu_Y + \big(1 + \lambda_v(t) g_{\mathcal{B}}(u)\big)\big(Y_x(u) - \mu_Y\big), 0, 1\right], \tag{13}$$

where μ_Y is the global mean luminance and $\lambda_v(t)$ is the dynamic shaping strength. Chrominance channels are kept unchanged, and the transformed luminance channel is converted back to RGB. The Gaussian field changes smoothly near object boundaries, and the zero-mean correction reduces global illumination drift.

3.4 Feature-Driven Curriculum Controller

A fixed shaping strength is suboptimal because representation geometry changes during training [16, 17]. We therefore modulate $\lambda_v(t)$ through a feature-state controller. At selected monitoring steps, we extract positive sample features $z_i^{(t)} \in \mathbb{R}^d$ from the last feature map before the detection head. We compute three geometry statistics: the foreground-background cosine margin on the low-LVI diagnostic set, the effective rank, and the pseudo-volume:

$$\begin{aligned} m_t^{\text{low}} &= \mathbb{E}_{i \in \mathcal{S}_t^{\text{low}}} \left[1 - \cos(z_i^{\text{fg}}, z_i^{\text{bg}})\right], \\ d_{\text{eff}}(t) &= \frac{(\text{tr}\Sigma_t)^2}{\text{tr}(\Sigma_t^2) + \epsilon}, \qquad v_t = \frac{1}{d} \log\det(\Sigma_t + \epsilon I), \end{aligned} \tag{14}$$

where Σ_t is the foreground feature covariance matrix. The low-LVI set $\mathcal{S}_t^{\text{low}}$ is defined by the LVI measurement in Eq. (11). All statistics are computed without gradient, smoothed with EMA, direction-normalized, and combined into a control state $C_t \in [0, 1]$. The shaping strength is updated as

$$\lambda_v(t) = g(t)\,\text{clip}\left(\lambda_0 + k(C_t - C^\star), \lambda_{\min}, \lambda_{\max}\right), \tag{15}$$

where $g(t)$ is a training-stage gate. This couples visibility shaping to the evolving representation state rather than to a fixed epoch-only schedule.

3.5 Training Instantiation

TANS is implemented as a plug-in operator for standard detection training pipelines. SPS is applied to object-level geometric composition, such as Mosaic and Copy-Paste, before final geometric validity checks. LVS is applied after geometric composition and before photometric normalization. The detector architecture, loss function, label assignment rule, and inference procedure are unchanged.

In our main implementation, source objects are selected from the training set. SPS computes the target log-area using Eq. (6) and rescales the object crop using Eq. (7). The resized object is pasted into the composed image under boundary and overlap constraints. After all geometric operations are completed, LVS constructs the box-conditioned Gaussian field and applies Eq. (13). Annotations are updated only for geometric transformations; visibility shaping leaves box coordinates and category labels unchanged.

For the attribute-regime visualizations, we use Mosaic as the reference baseline because it already provides strong scale, context, and composition variation. Improvements over Mosaic therefore indicate gains beyond generic augmentation strength.

4 Experiments

4.1 Experimental Settings

Datasets and detectors. We evaluate TANS on MS COCO [9] and Pascal VOC 2007+2012 [10]. COCO is the primary benchmark for final performance, representation probing, and schedule analysis. We use Faster R-CNN [2] with FPN under MMDetection [39] as the two-stage detector and YOLOv11 as the one-stage detector. Transformer-based detector results(RT-DETR[40]) are reported in Appendix F.

Metrics. For Faster R-CNN, we report AP, AP_{50}, AP_{75}, AP_s, AP_m, and AP_l. For YOLOv11, we report $\text{mAP}_{50:95}$, mAP_{50}, precision, and recall. For representation diagnostics, objects are assigned to scale and LVI bins as coordinates of the intrinsic attribute spectrum. These bins are used for diagnosis rather than for manually defining a hard-example mining objective.

Training protocol. Faster R-CNN on COCO follows the official MMDetection $1\times$ schedule. Faster R-CNN on Pascal VOC follows the $2\times$ schedule used in our implementation. Detailed learning-rate schedules, augmentation parameters, and resource usage are provided in Appendix G. For YOLOv11 on COCO, all methods use the same image size, batch setting, 1000-epoch maximum budget, and early stopping with patience 50. Each method reports its best validation checkpoint; TANS stops automatically at epoch 811 after 50 epochs without further improvement. Figs. 3

Table 1: **COCO results with Faster R-CNN under the official MMDetection $1\times$ schedule.** We highlight our proposed components in color. **Bold** numbers denote the best result, and underlined numbers denote the second best in each block.

Methods	MS COCO Metrics					
	AP ↑	AP_{50} ↑	AP_{75} ↑	AP_s ↑	AP_m ↑	AP_l ↑
Detector: Faster R-CNN (ResNet-50-FPN)						
Baseline	35.8	57.0	38.4	20.6	39.1	46.4
+ Mixup ($\alpha = 0.5$) [18]	35.9	57.3	38.4	20.8	39.2	46.1
+ Manifold Mixup ($\alpha = 1.0$) [41]	36.4	57.5	39.2	21.2	40.0	46.5
+ Cutout [27]	36.1	57.4	38.7	21.1	39.6	46.6
+ CutMix ($\alpha = 1.0$) [19]	36.6	58.0	39.2	21.4	39.9	47.0
+ CutMix + SUMix [42]	36.9	58.2	39.5	21.6	40.3	47.2
+ Simple Copy-Paste [20]	37.7	58.6	40.8	22.9	40.9	48.0
+ SPS-only (Step 1)	38.2	59.1	41.5	23.3	41.5	48.8
+ LVS-only (Step 2)	38.1	59.0	41.5	23.6	41.4	48.5
+ TANS (Step 1 & 2)	**39.0**	**60.0**	**42.7**	**24.7**	**42.2**	**49.7**
Detector: Faster R-CNN (ResNet-101-FPN)						
Baseline	37.7	59.0	40.7	21.6	41.4	49.2
+ CutMix ($\alpha = 1.0$) [19]	38.6	59.7	41.6	22.4	42.5	49.9
+ CutMix + SUMix [42]	38.9	60.0	42.6	22.8	42.9	50.1
+ Simple Copy-Paste [20]	39.9	60.3	43.6	24.4	43.8	51.0
+ SPS-only (Step 1)	40.5	60.6	44.3	24.4	44.6	52.2
+ LVS-only (Step 2)	40.7	60.8	44.4	25.1	44.6	51.9
+ TANS (Step 1 & 2)	**41.8**	**61.5**	**45.5**	**26.1**	**45.9**	**52.7**

Table 2: **Results on Pascal VOC and YOLOv11.** Left: Faster R-CNN on Pascal VOC under the $2\times$ schedule used in our implementation. Right: YOLOv11 on COCO under a common 1000-epoch maximum budget with early stopping. Values are reported in percentage points.

(a) Faster R-CNN on Pascal VOC

Methods	AP ↑
ResNet-50	
No Aug.	79.8
Copy-Paste	81.2
SPS-only (Step 1)	81.9
LVS-only (Step 2)	81.7
TANS (Step 1 & 2)	**82.6**
ResNet-101	
No Aug.	81.1
Copy-Paste	82.3
SPS-only (Step 1)	82.7
LVS-only (Step 2)	82.5
TANS (Step 1 & 2)	**83.5**

(b) YOLOv11 on COCO

Methods	$mAP_{50:95}$ ↑	mAP_{50} ↑	Prec. ↑	Rec. ↑
No Aug.	36.1	51.5	61.5	47.5
Mixup	40.7	56.9	63.8	52.1
Mosaic	43.0	59.4	66.9	54.4
Mosaic + Mixup	45.2	61.9	70.0	56.9
OA-DG [31]	43.6	60.1	69.1	55.3
ControlAug [29]	44.8	61.6	70.8	56.1
SPS-only (Step 1)	45.5	62.2	**71.8**	56.2
LVS-only (Step 2)	45.7	62.4	70.6	56.6
TANS (Step 1 & 2)	**46.6**	**63.4**	71.4	**58.3**

and 4 use a common 500-epoch diagnostic window to compare representation trajectories before the policies terminate at different epochs.

4.2 Main Results

Table 1 reports COCO results with Faster R-CNN under the official MMDetection $1\times$ schedule. SPS-only isolates the geometric scale-prior branch, LVS-only isolates the local-visibility branch, and TANS combines both operators. Across ResNet-50-FPN and ResNet-101-FPN, TANS improves over strong attribute-blind vicinal baselines, indicating that the gain is not explained by generic augmentation strength alone.

Table 2 further evaluates transfer across datasets and detector families. On Pascal VOC, TANS improves over Copy-Paste under the Faster R-CNN $2\times$ schedule. On COCO with YOLOv11, TANS

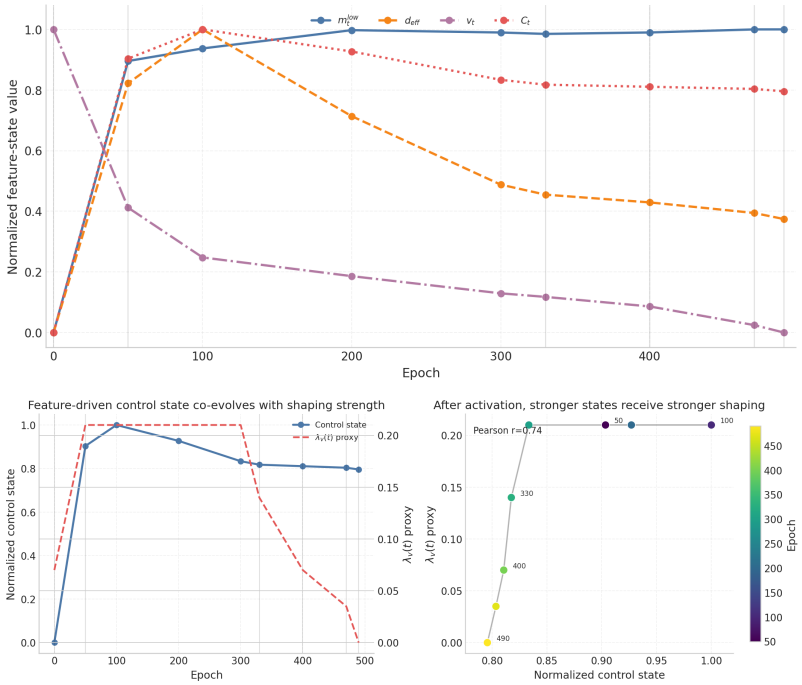


Figure 3: **Feature-state dynamics under a controlled 500-epoch diagnostic window.** Top: foreground-background margin, effective rank, pseudo-volume, and the derived control state from saved checkpoints. Bottom: the controller couples shaping strength to the representation state. These dynamics indicate that TANS is representation-aligned nuisance shaping rather than a fixed-strength augmentation policy.

improves over the Mosaic baseline from 43.0 to 46.6 $\text{mAP}_{50:95}$, yielding a 3.6-point gain while also improving mAP_{50}, precision, and recall. Attribute-regime heatmaps and attribute-sliced recall dynamics are provided in Appendix A; they are used to diagnose attribute dependence rather than to define TANS as hard-example mining.

For YOLOv11, "SPS-only" denotes Mosaic+SPS: SPS replaces Mosaic's random scale proposal with class-conditional scale-prior shaping. Direct VOC comparisons and five-run statistics are reported in Appendix F.

4.3 Feature-State Dynamics and Schedule Stability

Beyond aggregate AP, we examine whether shaping is aligned with the evolving representation geometry. Fig. 3 tracks the foreground-background margin, effective rank, pseudo-volume, and derived control state within the controlled 500-epoch diagnostic window. These quantities evolve nonlinearly, showing that intrinsic attributes interact with a moving feature manifold rather than acting as static dataset metadata. The controller therefore updates shaping strength according to representation state instead of relying only on a fixed epoch index.

Table 3 distinguishes feature-state feedback from scale-aware policy search. Fig. 4 compares fixed and adaptive schedules in the same diagnostic window. This analysis is separated from the benchmark tables: the tables report best converged checkpoints under a common protocol, while the figure isolates how schedule design interacts with representation dynamics. The controller-based schedule maintains lower tail variation and better late-stage stability than heuristic schedules, supporting feature-state feedback for shaping the vicinal distribution.

4.4 Representation-Space Probe

We further probe the detection-head logit space to test whether TANS changes class-level separability. For each ground-truth object, we compute the GT class-logit margin, defined as the logit of the correct class minus the largest competing logit, and the GT class top-1 accuracy. As shown in Fig. 5,

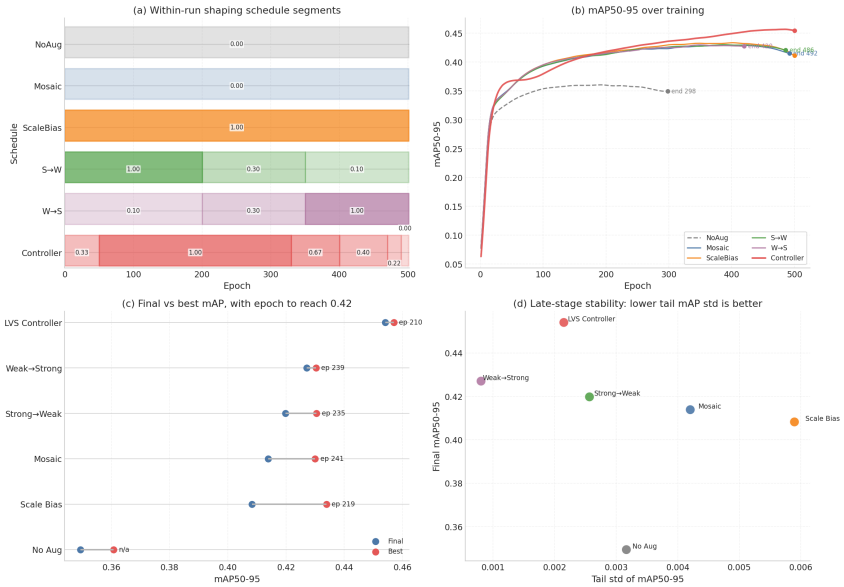


Figure 4: **Schedule comparison in a controlled 500-epoch diagnostic window.** (a) Shaping-strength segments. (b) $\text{mAP}_{50:95}$ trajectories. (c) Final vs. best mAP. (d) Late-stage stability measured by tail mAP standard deviation. Fixed and adaptive schedules use the same SPS+LVS augmentation content. The purpose of this figure is to compare how these schedules interact with representation geometry; final benchmark numbers are reported from converged checkpoints in Tables 1 and 2.

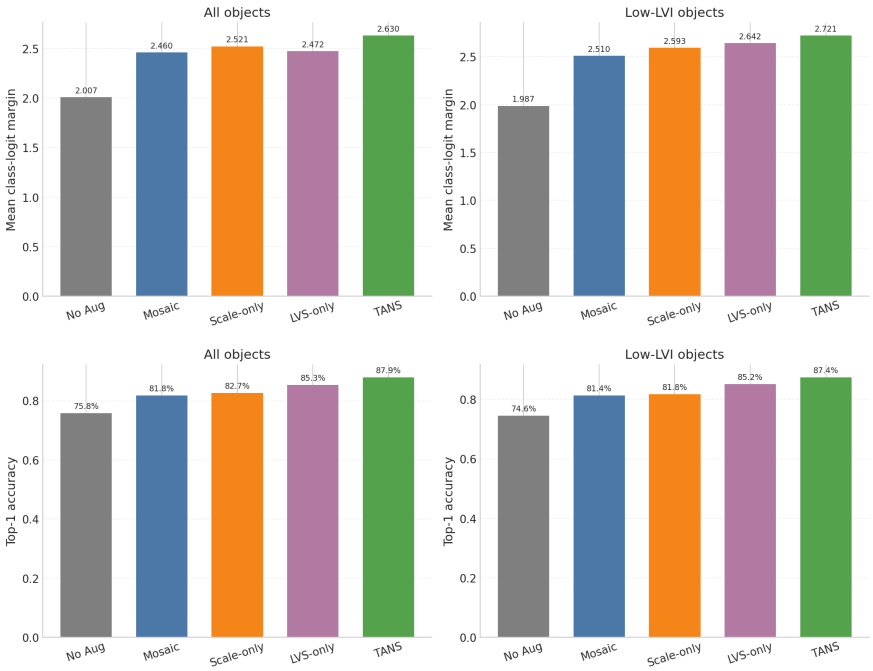


Figure 5: **Detection-head logit-space probes.** Top: GT class-logit margin, where higher values indicate stronger separation from competing classes. Bottom: GT class top-1 accuracy in logit space. TANS improves logit-space separability over Mosaic and the single-factor variants, supporting the claim that target-aware shaping reduces attribute-conditioned representation ambiguity.

TANS produces the strongest margin and top-1 accuracy on both the full object set and the low-LVI diagnostic slice. This suggests that the improvement is associated with reduced attribute-conditioned ambiguity in the detector head, not merely with a larger number of final detections.

Table 3: **Scale-aware policy search and target-aware shaping.**

Dimension	Scale-Aware Automatic Augmentations [43]	TANS
Optimization paradigm	Searches image- and instance-level scale-aware transformation policies.	Class-conditional scale priors and box-conditioned visibility shaping.
Sample generation	Applies searched transformations, probabilities, and magnitudes.	Monotone log-scale transport; local contrast modulation preserves geometry.
Controller	Searched policies; heuristic image-level and differentiable instance-level adaptation.	Measured feature-state statistics govern shaping strength.
Interpretability	Selected operators, probabilities, magnitudes, and schedules.	Scale priors, local visibility, and measured representation geometry.

4.5 Attribute Dependence under Scale Shifts

To test whether scale-prior shaping improves robustness to changes in object scale, we directly evaluate trained YOLOv11 detectors on perturbed COCO validation images, without additional training or fine-tuning for these scale-shift conditions. We randomly select 20%, 30%, or 40% of non-crowd objects and expand them by factors in $[1.5, 2.0]$ or shrink them by factors in $[0.1, 0.5]$. Categories and object centers are retained, and unselected objects remain unchanged. All methods use identical perturbed images, annotations, and evaluation settings.

Table 4: **Scale-shift evaluation on COCO with YOLOv11.** mAP is reported on the $[0, 1]$ scale. Percentages indicate the fraction of objects perturbed. Mosaic+SPS isolates the scale-prior branch.

		Expansion			Shrinkage		
Method	Clean	20%	30%	40%	20%	30%	40%
No Aug.	0.3611	0.3665	0.3666	0.3659	0.2901	0.2551	0.2248
Mosaic	0.4304	0.4326	0.4336	0.4326	0.3522	0.3135	0.2780
Mosaic+SPS	**0.4554**	**0.4596**	**0.4595**	**0.4601**	**0.4066**	**0.3751**	**0.3494**

Table 4 shows that shrinkage substantially reduces detection accuracy, whereas expansion has a smaller effect. The advantage of Mosaic+SPS over Mosaic grows from 2.50 percentage points on clean images to 5.44, 6.16, and 7.14 points as shrinkage affects more objects. Thus, the scale-prior branch strengthens performance precisely when object scales move toward the more difficult small-scale regime, complementing the clean-set gains with direct evidence under controlled scale shifts.

The original scale diagnostic in Appendix A.3, Fig. S3(c), also shows reduced class-level SDE–AP correlation on COCO: SPS reaches 0.421/0.494 for ResNet-50/101, versus 0.503/0.566 without augmentation.

5 Conclusion and Limitations

We propose Target-Aware Nuisance Shaping (TANS) to mitigate intrinsic attribute biases in object detection by reformulating data augmentation as an object-conditioned vicinal distribution design. By dynamically adapting shaping strength to the evolving representation geometry, TANS improves performance on MS COCO and Pascal VOC while reducing attribute entanglement. Current limitations include the luminance-based visibility proxy, composition-induced context changes, training overhead, and dependence on bounding-box annotations. Future work will extend this framework to broader nuisance factors, including pose and occlusion, with reduced annotation dependency.

The supplement provides attribute-regime and recall analyses (Appendix A), scale-distribution entropy (Appendix B), algorithms (Appendix C), correlation analyses (Appendix D), methodological details (Appendix E), additional detector comparisons and repeated-run results (Appendix F), and implementation settings and resources (Appendix G).

Acknowledgments and Disclosures

This work is supported by the National Natural Science Foundation of China (No.62371356 and No.62405238), in part by China Postdoctoral Science Foundation (No.2024M752512). The authors declare no competing interests.

References

[1] R. Girshick, "Fast r-cnn," *arXiv preprint arXiv:1504.08083*, 2015.

[2] S. Ren, "Faster r-cnn: Towards real-time object detection with region proposal networks," *arXiv preprint arXiv:1506.01497*, 2015.

[3] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. Springer, 2016, pp. 21–37.

[4] J. Redmon, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[5] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7263–7271.

[6] Z. Zhang, T. He, H. Zhang, Z. Zhang, J. Xie, and M. Li, "Bag of freebies for training object detection neural networks," *arXiv preprint arXiv:1902.04103*, 2019.

[7] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[9] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. Springer, 2014, pp. 740–755.

[10] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, pp. 303–338, 2010.

[11] L. Herranz, S. Jiang, and X. Li, "Scene recognition with cnns: objects, scales and dataset bias," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 571–579.

[12] B. Krawczyk, "Learning from imbalanced data: open challenges and future directions," *Progress in artificial intelligence*, vol. 5, no. 4, pp. 221–232, 2016.

[13] K. Oksuz, B. C. Cam, S. Kalkan, and E. Akbas, "Imbalance problems in object detection: A review," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 10, pp. 3388–3415, 2020.

[14] Z. Liu, Z. Miao, X. Zhan, J. Wang, B. Gong, and S. X. Yu, "Large-scale long-tailed recognition in an open world," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 2537–2546.

[15] J. Tan, C. Wang, B. Li, Q. Li, W. Ouyang, C. Yin, and J. Yan, "Equalization loss for long-tailed object recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 662–11 671.

[16] B. Kang, S. Xie, M. Rohrbach, Z. Yan, A. Gordo, J. Feng, and Y. Kalantidis, "Decoupling representation and classifier for long-tailed recognition," *arXiv preprint arXiv:1910.09217*, 2019.

[17] Y. Ma, L. Jiao, F. Liu, S. Yang, X. Liu, and L. Li, "Curvature-balanced feature manifold learning for long-tailed classification," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 15 824–15 835.

[18] H. Zhang, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[19] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "Cutmix: Regularization strategy to train strong classifiers with localizable features," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6023–6032.

[20] G. Ghiasi, Y. Cui, A. Srinivas, R. Qian, T.-Y. Lin, E. D. Cubuk, Q. V. Le, and B. Zoph, "Simple copy-paste is a strong data augmentation method for instance segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 2918–2928.

[21] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of big data*, vol. 6, no. 1, pp. 1–48, 2019.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[23] S.-A. Rebuffi, S. Gowal, D. A. Calian, F. Stimberg, O. Wiles, and T. A. Mann, "Data augmentation can improve robustness," *Advances in Neural Information Processing Systems*, vol. 34, pp. 29 935–29 948, 2021.

[24] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, 2020, pp. 13 001–13 008.

[25] R. Takahashi, T. Matsubara, and K. Uehara, "Data augmentation using random image cropping and patching for deep cnns," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 9, pp. 2917–2931, 2019.

[26] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "Dropblock: A regularization method for convolutional networks," *Advances in neural information processing systems*, vol. 31, 2018.

[27] T. DeVries, "Improved regularization of convolutional neural networks with cutout," *arXiv preprint arXiv:1708.04552*, 2017.

[28] Y. Chen, Y. Li, T. Kong, L. Qi, R. Chu, L. Li, and J. Jia, "Scale-aware automatic augmentation for object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 9563–9572.

[29] H. Fang, B. Han, S. Zhang, S. Zhou, C. Hu, and W.-M. Ye, "Data augmentation for object detection via controllable diffusion models," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2024, pp. 1257–1266.

[30] H. Zhu, T. Pan, R. Qin, J.-H. Yong, and B. Wang, "Recon: Region-controllable data augmentation with rectification and alignment for object detection," *arXiv preprint arXiv:2510.15783*, 2025.

[31] W. Lee, D. Hong, H. Lim, and H. Myung, "Object-aware domain generalization for object detection," in *proceedings of the AAAI conference on artificial intelligence*, vol. 38, no. 4, 2024, pp. 2947–2955.

[32] T. Çelik and T. Tjahjadi, "Automatic image equalization and contrast enhancement using Gaussian mixture modeling," *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 145–156, 2012.

[33] Z. Lin, C. Liu, W. Qi, and S. C. Chan, "A color/illuminance aware data augmentation and style adaptation approach to person re-identification," *IEEE Access*, vol. 9, pp. 115 826–115 838, 2021.

[34] W. Zhang, Y. Kinoshita, and H. Kiya, "Image-enhancement-based data augmentation for improving deep learning in image classification problem," in *2020 IEEE International Conference on Consumer Electronics–Taiwan (ICCE-Taiwan)*, 2020, pp. 1–2.

[35] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on knowledge and data engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[36] A. Estabrooks, T. Jo, and N. Japkowicz, "A multiple resampling method for learning from imbalanced data sets," *Computational intelligence*, vol. 20, no. 1, pp. 18–36, 2004.

[37] G. James, D. Witten, T. Hastie, R. Tibshirani, and J. Taylor, "Resampling methods," in *An Introduction to Statistical Learning: with Applications in Python*. Springer, 2023, pp. 201–228.

[38] D. A. Reynolds *et al.*, "Gaussian mixture models," *Encyclopedia of biometrics*, vol. 741, no. 659-663, 2009.

[39] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu *et al.*, "Mmdetection: Open mmlab detection toolbox and benchmark," *arXiv preprint arXiv:1906.07155*, 2019.

[40] Y. Zhao, W. Lv, S. Xu, J. Wei, G. Wang, Q. Dang, Y. Liu, and J. Chen, "Detrs beat yolos on real-time object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 16 965–16 974.

[41] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, and Y. Bengio, "Manifold mixup: Better representations by interpolating hidden states," in *International conference on machine learning*. PMLR, 2019, pp. 6438–6447.

[42] H. Qin, X. Jin, H. Zhu, H. Liao, M. A. El-Yacoubi, and X. Gao, "Sumix: Mixup with semantic and uncertain information," in *European Conference on Computer Vision*. Springer, 2024, pp. 70–88.

[43] Y. Chen, P. Zhang, T. Kong, Y. Li, X. Zhang, L. Qi, J. Sun, and J. Jia, "Scale-aware automatic augmentations for object detection with dynamic training," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 2, pp. 2367–2383, 2023.

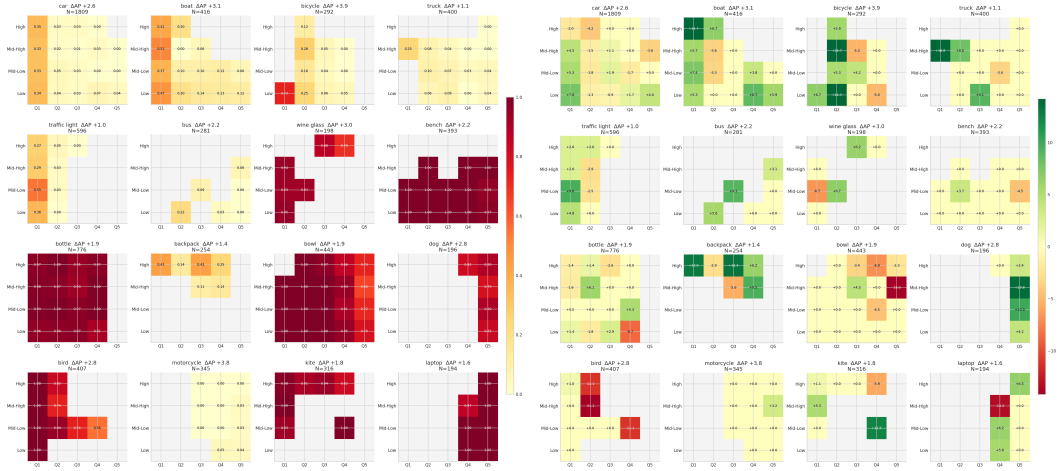
[44] P. Sedgwick, "Pearson's correlation coefficient," *Bmj*, vol. 345, 2012.

Supplementary Material

A Additional Analysis

A.1 Attribute-Regime Bias and TANS-Induced Redistribution

Standard AP averages over the full object population and can hide how detector behavior varies across the intrinsic attribute spectrum. We therefore analyze each ground-truth object according to its scale bin and LVI bin. Fig. S1 reports selected-class heatmaps. The left panel shows the Mosaic baseline miss-rate structure, while the right panel shows the TANS gain over Mosaic in the same LVI $\times$ scale bins. We retain neutral and negative cells rather than filtering the visualization, because the goal is to reveal how model behavior is redistributed over attribute states rather than to present only favorable cases.



(a) Mosaic baseline miss-rate. (b) TANS gain over Mosaic.

Figure S1: **Attribute-regime bias and TANS-induced redistribution.** Each heatmap is computed over LVI quantiles and scale quantiles, where rows indicate local visibility from Low to High and columns indicate scale bins from Q1 to Q5. **Left:** Mosaic baseline miss-rate, showing that residual errors remain organized by intrinsic attributes even under a strong attribute-blind augmentation baseline. **Right:** TANS$-$Mosaic gain in Recall@0.5, measured in percentage points. The non-uniform pattern indicates that TANS changes the detector's dependence on intrinsic attribute states rather than applying a uniform performance shift.

A.2 Attribute-Sliced Recall Dynamics

Fig. S2 tracks Recall@0.5 over training for three diagnostic slices of the attribute spectrum: low-LVI objects, small-scale objects, and objects that are simultaneously low-LVI and small-scale. These slices are used to observe attribute dependence during learning; they should not be interpreted as the full problem definition. TANS provides more favorable learning dynamics on the scale-related and joint attribute slices while remaining competitive on the visibility slice, suggesting that SPS and LVS act complementarily.

A.3 Auxiliary Scale Diagnostic with SDE

We retain Scale Diversity Entropy (SDE) as an auxiliary diagnostic for the scale branch. SDE is useful for measuring class-level scale diversity and for showing that scale-aware shaping can weaken the dependence between class-level scale structure and detection performance. In the current TANS formulation, however, SDE is not the main optimization target; it serves as supporting evidence for the geometric effectiveness of SPS.

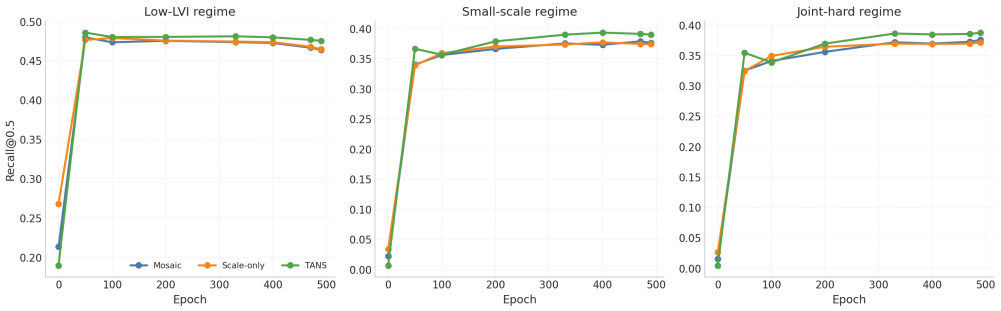


Figure S2: **Attribute-sliced recall dynamics.** We compare Mosaic, SPS-only, and TANS across low-LVI, small-scale, and joint attribute slices. The curves are used to diagnose how representation learning evolves over intrinsic attributes, rather than to define TANS as a selected-sample mining method.

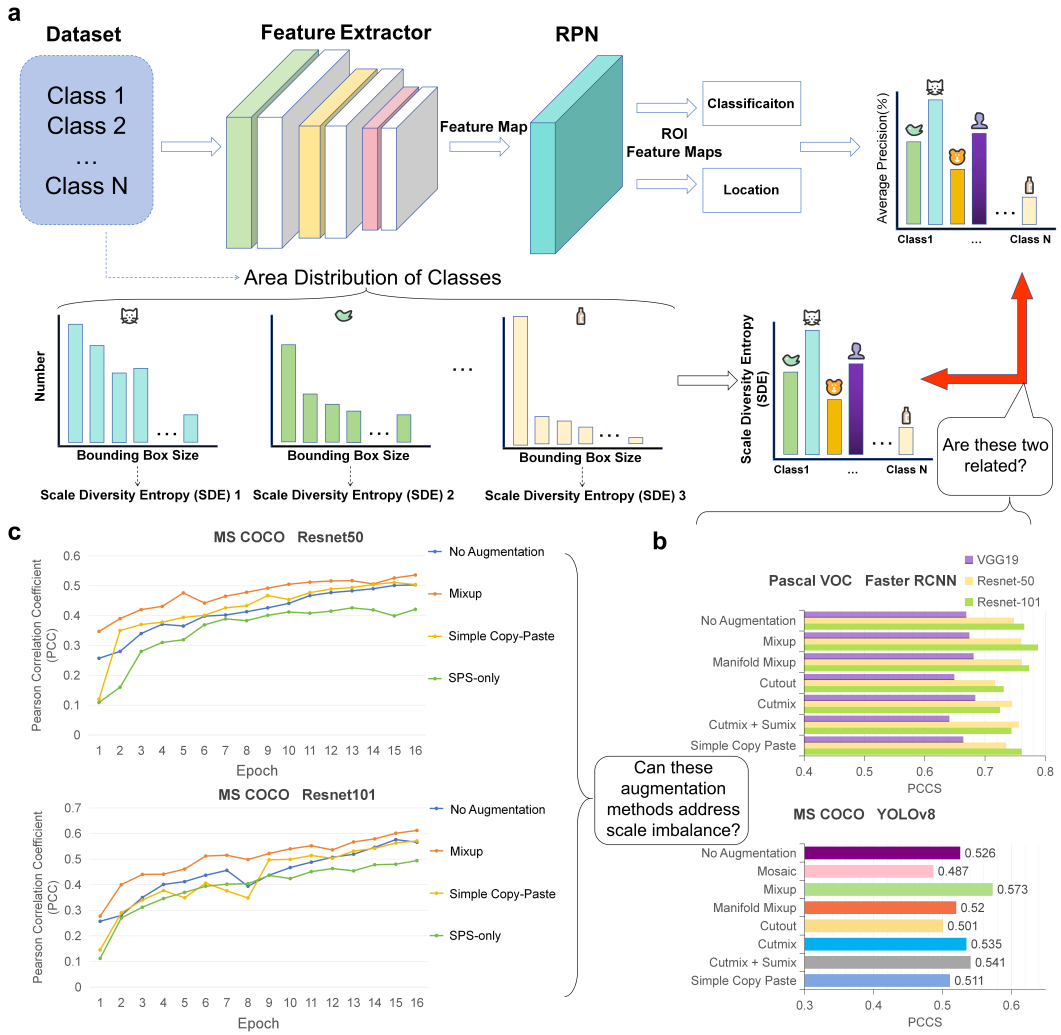


Figure S3: **Auxiliary scale-only diagnostic based on Scale Diversity Entropy (SDE). a** Conceptual framework for analyzing the correlation between class-level SDE and detection AP. **b** Pearson correlation coefficients between SDE and AP under different datasets, detectors, backbones, and augmentation methods. **c** Training dynamics of the SDE–AP correlation on COCO with ResNet-50 and ResNet-101, including the SPS-only scale branch. This figure supports the geometric transport branch of our framework. Note: "SPS-only" targets this specific scale imbalance.

B Entropy of Object Scale Distributions

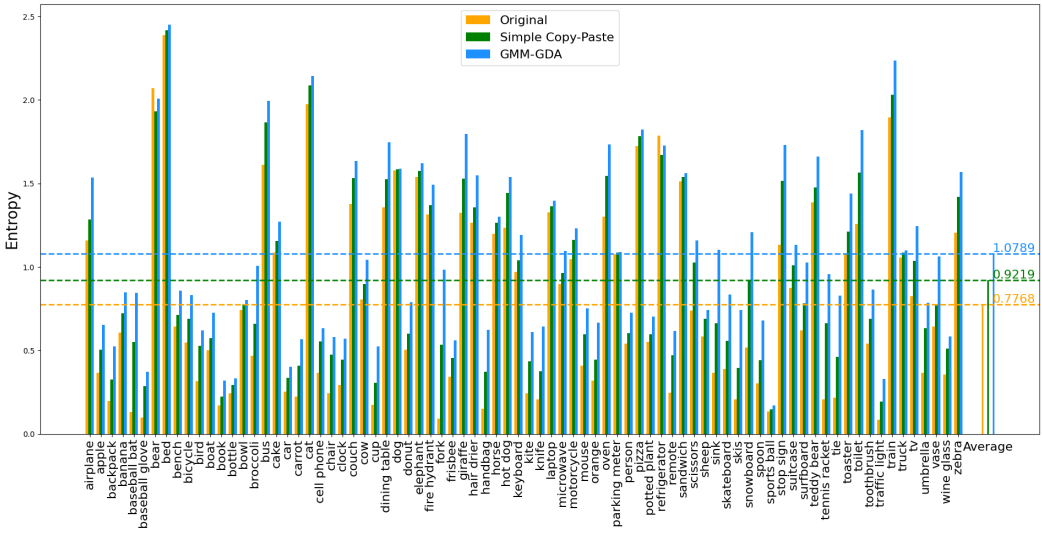


Figure S4: **Entropy of Object Scale Distributions Across 80 COCO Classes [9].** This figure illustrates the entropy of object scale distributions for each of the 80 categories in the COCO dataset under three conditions: the original dataset, after applying Simple Copy-Paste, and after employing SPS-only. Each bar represents the entropy of the bounding-box area distribution within a given class. The categories are sorted alphabetically along the x-axis. At the far right of the figure, the average entropy of the 80 classes for each of the three conditions is marked above the dashed lines of the corresponding color.

Scale bias severity is inversely related to SDE. When the distribution is more extreme, the SDE is lower, whereas more balanced distributions have higher SDE. The observed increase in SDE when using SPS-only highlights its efficacy in promoting balanced object size distributions.

C Algorithmic Implementations

This section details the pseudo-code for the proposed modules. Algorithm S1 outlines the Scale Prior Shaping (SPS) process for geometric transport. Algorithm S2 describes the Local Visibility Shaping (LVS) process for photometric modulation. Algorithm S3 illustrates how these operators are serially integrated with the feature-driven controller.

D Pearson Correlation Coefficients Analysis

Our quantitative framework uses Scale Diversity Entropy to measure scale distribution structure and Pearson correlation coefficients [44] to assess its relationship with detection performance. In Appendix B, we compare SDE for the original dataset, Simple Copy-Paste, and SPS-enhanced data. The results show that SPS increases scale diversity and weakens the dependence between class-level scale structure and detection performance.

E Extended Methodological Details

Zero-mean field normalization. On the image domain Ω, we normalize the fused field to $[-1, 1]$ to reduce illumination drift during Eq. (13):

$$g_{\mathcal{B}}(u) = \frac{w_{\mathcal{B}}(u) - \frac{1}{|\Omega|}\sum_{v\in\Omega} w_{\mathcal{B}}(v)}{\max_{v\in\Omega}\left|w_{\mathcal{B}}(v) - \frac{1}{|\Omega|}\sum_{u\in\Omega} w_{\mathcal{B}}(u)\right| + \epsilon}, \tag{S1}$$

Algorithm S1 Scale Prior Shaping (SPS) Integration

Input: Dataset $\mathcal{D}$, class c, bounding-box log-areas $\mathcal{A}_c$, clipped support $[Q_\eta^{(c)}, Q_{1-\eta}^{(c)}]$
Output: Remapped scale factor s_i for object i
Offline prior construction (per class c):
$K_c \leftarrow \arg\min_{K\in\{1,\dots,10\}}\{\mathrm{BIC}_c(K)\}$
Fit GMM $p_a^{(c)}(a) = \sum_{k=1}^{K_c} \pi_k \mathcal{N}(a \mid \mu_k, \sigma_k^2)$
Construct $q_a^{(c)}(a) \propto [\max(p_a^{(c)}) - p_a^{(c)}(a)]_+ + 10^{-6}$ inside the clipped support
Compute cumulative distribution functions $F_p^{(c)}$ and $F_q^{(c)}$
Online remapping (for object i of class c):
Retrieve original log-area a_i
Compute target log-area $a_i^\star = \left(F_q^{(c)}\right)^{-1}\left(F_p^{(c)}(a_i)\right)$
Calculate isotropic scale factor $s_i = \sqrt{\exp(a_i^\star - a_i)}$
Clip s_i to $[s_{\min}, s_{\max}] = [0.1, 2.0]$
Apply boundary clipping and reject invalid candidates with IoU > 0.7 or area < 16 pixels^2
Return s_i

Algorithm S2 Local Visibility Shaping (LVS) Integration

Input: Augmented image X with luminance Y_x, bounding boxes $\mathcal{B}$, shaping strength $\lambda_v(t)$
Output: Photometrically modulated image X'
for each box $b_i \in \mathcal{B}$ **do**
    Compute the LVI measurement using the foreground box and equal-area local background ring
    Construct Gaussian field $G_i(u)$ centered at b_i
end for
Fuse fields uniformly: $w_{\mathcal{B}}(u) = 1 - \prod_i \left(1 - G_i(u)\right)$
Compute image-level mean $\mu_w = \frac{1}{|\Omega|}\sum_u w_{\mathcal{B}}(u)$
Convert to zero-mean field: $g_{\mathcal{B}}(u) = \left(w_{\mathcal{B}}(u) - \mu_w\right)/\text{NormFactor}$
Calculate global mean luminance μ_Y
Apply $Y'_x(u) = \mathrm{clip}\left(\mu_Y + (1 + \lambda_v(t) g_{\mathcal{B}}(u))(Y_x(u) - \mu_Y), 0, 1\right)$
Reconstruct RGB image X' using transformed luminance Y'_x
Return X'

Algorithm S3 Target-Aware Nuisance Shaping (TANS) Pipeline

Input: Training iteration t, base image IM_{main}, source objects $\{o_j\}$
Controller monitoring:
if the current epoch is a monitoring epoch **then**
    Extract feature statistics Σ_t, z_i^{fg}, and z_i^{bg} from the last feature map before the detection head
    Update C_t and calculate $\lambda_v(t)$ using Eq. 15
end if
Geometric transport (SPS):
for each object o_j **do**
    Obtain scale factor $s_j \leftarrow \mathrm{SPS}(o_j)$ via Algorithm S1
    Resize o_j using s_j and perform boundary, area, and IoU validity checks
end for
$IM_{\text{geo}}, \mathcal{B}_{\text{geo}} \leftarrow \mathrm{GeometricComposition}(IM_{\text{main}}, \{o_j\})$
Photometric modulation (LVS):
$IM_{\text{final}} \leftarrow \mathrm{LVS}(IM_{\text{geo}}, \mathcal{B}_{\text{geo}}, \lambda_v(t))$ via Algorithm S2
Return $(IM_{\text{final}}, \mathcal{B}_{\text{geo}})$

Feature statistics. The controller monitors the foreground-background margin, effective rank, and pseudo-volume without gradient. These statistics are smoothed over monitoring steps and normalized before being averaged into the control state. The measurement phase is decoupled from back-

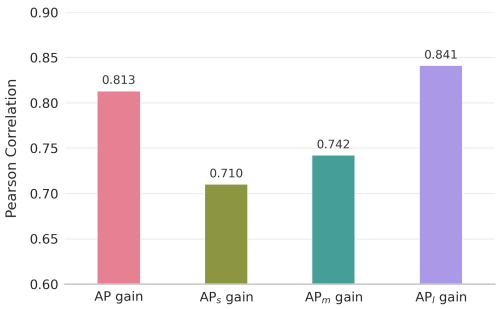


Figure S5: PCCs between change in SDE and performance gains across different object scales. From left to right: all, small, medium, and large scale.

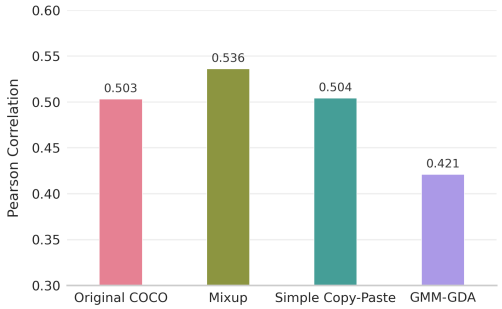


Figure S6: PCCs between detection accuracy and average SDE of COCO under different augmentation methods.

propagation, so the controller changes only the subsequent augmentation strength and does not introduce an additional training loss.

F Additional Detector Comparisons

Direct comparisons and repeated runs. Table S1 compares TANS with scale-aware policy search (SA-AutoAug) and region-controllable augmentation (ReCon) using Faster R-CNN on Pascal VOC. The five-run TANS means exceed SA-AutoAug by 0.92/1.07 AP points and ReCon by 1.62/1.47 points for ResNet-50/101, respectively. The small run-to-run standard deviations support the repeatability of the gains.

Table S1: **Direct Pascal VOC comparisons with Faster R-CNN.** TANS reports mean $\pm$ standard deviation over five independent training runs. The original single-run results are retained in Table 2.

Method	ResNet-50 AP	ResNet-101 AP
ReCon [30]	80.9	82.0
SA-AutoAug [28]	81.6	82.4
TANS (five runs)	$\mathbf{82.52 \pm 0.17}$	$\mathbf{83.47 \pm 0.23}$

Transformer-based detectors. To examine whether the proposed shaping strategy transfers beyond CNN-style detectors, we evaluate the scale-prior branch on RT-DETR[40] with a ResNet-18 backbone. We use the official RT-DETR recipe with batch size 4, 72 training epochs, initial learning rate 1×10^{-4}, weight decay 1×10^{-4}, AdamW optimizer, and eight RTX 4090 GPUs. The RT-DETR baseline already includes common augmentations, including RandomPhotometricDistort, RandomZoomOut, RandomIoUCrop, and MultiScaleInput.

Table S2: RT-DETR results on COCO.

Method	**AP**	**AP$_{50}$**	**AP$_{75}$**	**AP$_s$**	**AP$_m$**	**AP$_l$**
RT-DETR baseline	46.3	63.4	50.3	28.3	49.7	62.5
SPS-only	46.9	63.9	51.1	28.7	50.3	63.3
LVS-only	47.2	64.1	51.5	29.0	50.6	63.5
TANS	48.0	64.8	52.4	29.9	51.4	64.2

G Parameter Settings and Compute Resources

This section summarizes the implementation details used in the main experiments. TANS is used only during training. It does not change the detector architecture, loss function, label assignment rule, or inference procedure.

Training protocols. For Faster R-CNN on COCO, we follow the official MMDetection $1\times$ schedule. For Faster R-CNN on Pascal VOC, we use the $2\times$ schedule in our implementation on the VOC 2007+2012 training split. The detailed schedule choice is therefore tied to the corresponding official or implementation recipe rather than to an additional long-horizon training setting.

For YOLOv11 on COCO, all methods use image size 640×640, per-GPU batch size 64, two RTX 4090 GPUs, the same 1000-epoch maximum budget, and early stopping with patience 50. Each method reports its best validation checkpoint; the TANS run stops automatically at epoch 811 after 50 epochs without further improvement. Standard HSV augmentation uses hue, saturation, and value settings 0.015, 0.7, and 0.4, respectively. Under this setting, the Mosaic baseline takes approximately 1 min 40 s per epoch, while TANS takes approximately 1 min 50 s per epoch. The complete TANS run takes about 25 hours. SPS prior fitting is performed offline once before training; LVS and the controller add only training-time overhead and incur no inference-time cost.

SPS parameters. For SPS, the maximum number of GMM components is $K_{\max} = 10$, and the component number K_c for each class is selected by BIC. The quantile clipping ratio is $\eta = 0.01$, and the smoothing constant in the reference prior is $\epsilon_q = 10^{-6}$. The scale factor is clipped to $[s_{\min}, s_{\max}] = [0.1, 2.0]$, matching the Simple Copy-Paste setting. A pasted candidate is rejected if its IoU with existing objects exceeds $\theta_{\text{iou}} = 0.7$ or if its final area is smaller than $A_{\min} = 16$ pixels^2. Boxes crossing image boundaries are clipped to the image region, and each candidate is retried at most five times.

LVI and LVS parameters. For the foreground luminance measurement in LVI, the Gaussian weight uses $\sigma_x = \kappa_{\text{fg}} w_i$ and $\sigma_y = \kappa_{\text{fg}} h_i$ with $\kappa_{\text{fg}} = 0.25$. The local background ring is constructed to have the same area as the foreground box. Concretely, the outer box has the same center as the foreground box and side lengths $\sqrt{2}w_i$ and $\sqrt{2}h_i$, and the foreground box is removed from this outer region. Equivalently, if dilation is parameterized as total side-length expansion, the expansion ratio is $\sqrt{2} - 1$; if it is parameterized as a per-side margin ratio, the value is $(\sqrt{2} - 1)/2$. This dilation parameter controls the background ring and should not be confused with κ_{fg}, which controls the Gaussian foreground weighting.

LVI is a measurement-derived visibility signal, not a statistical prior like the class-conditional scale prior in SPS. In the main implementation, it is used to measure the initial local visibility state of annotated objects and to form the low-LVI diagnostic set used by the feature-state controller. We do not introduce an additional hand-crafted LVI priority weight in the LVS field fusion. All annotated boxes contribute uniformly to the field, which corresponds to setting $\alpha_i = 1$ in an optional weighted formulation. The fused field is then zero-mean normalized by the maximum absolute deviation as in Eq. (S1). The active visibility shaping strength is clipped to $[0.3, 0.7]$, giving a local contrast factor $1 + \lambda_v(t) g_{\mathcal{B}}(u)$ in $[0.3, 1.7]$.

Feature-state controller. The controller monitors the feature state every 10 epochs. After a monitoring step, the updated shaping strength is used in the subsequent normal training process. Object features are extracted from the last feature map before the detection head. Foreground features are pooled from the ground-truth box region, while background features are pooled from the equal-area local background ring excluding pixels covered by other annotated boxes. The three controller signals, foreground-background margin, effective rank, and pseudo-volume, are combined with equal weights $(1/3, 1/3, 1/3)$.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction (§1) clearly state the core contributions of Target-Aware Nuisance Shaping (TANS), including Scale Prior Shaping (SPS) and Local Visibility Shaping (LVS), which are validated through extensive experiments in §4.

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations regarding the framework's focus on only scale and local visibility (ignoring factors like pose or occlusion), moderate computational overhead, and reliance on precise bounding box annotations are explicitly discussed in §5.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper proposes empirical data augmentation and representation learning methods. It does not contain formal theoretical proofs or theorems.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The experimental settings, including datasets and detectors, are specified in §4.1. Detailed algorithmic implementations and pseudo-code are provided in Appendix C.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: We plan to release the source code and configuration files upon publication of the paper.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The datasets, baseline detectors (Faster R-CNN, YOLOv11), and evaluation metrics are detailed in §4.1.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Following standard community practice in object detection on large-scale datasets like MS COCO and Pascal VOC, we report the standard aggregate metrics (AP, mAP) over the entire validation sets rather than error bars. However, we provide late-stage stability analysis via tail mAP standard deviation in Fig. 4. Furthermore, due to the high computational cost of training large-scale object detectors, we did not perform multiple independent runs to include error bars in the tables.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Detailed information regarding the computational resources, including hardware specifications, type of compute workers, and total experimental execution time, is provided in Appendix G.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: The research focuses on fundamental data augmentation for object detection and strictly adheres to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [No]

Justification: This work focuses on foundational data augmentation and representation learning for object detection. It does not have direct foreseeable negative societal impacts beyond the general implications of improving computer vision models.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer:[N/A]

Justification: The paper introduces a training-time data augmentation framework and does not release standalone models or datasets with a high risk for misuse.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Standard public datasets (MS COCO, Pascal VOC) and baseline detector architectures are appropriately cited in §4.1.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not introduce new datasets or large-scale foundational assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The research does not involve crowdsourcing or human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

Justification: The research does not involve human subjects and therefore does not require IRB approval.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development in this research does not involve Large Language Models.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.