INTERACTION STEERING FOR COMPOSITIONAL ROBUSTNESS IN VISION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Modern vision models are trained through outcome supervision: the predicted class should match the label. However, this supervision does not specify how visual factors should contribute jointly, allowing models to rely on spurious context or overlook useful combinations of object parts. We introduce *Interaction Steering*, a framework for mechanism supervision in vision models. Interaction Steering applies finite-difference operators to counterfactual edits of a model's score, isolating interactions such as object–background, object–co-occurring-object, and part–part effects. These operators audit how named visual factors jointly affect the score and define auxiliary steering and guard losses that move their interactions toward specified targets. Counterfactual edits are used only for these auxiliary losses, while the primary supervised objective remains on natural images. On Waterbirds, Interaction Steering increases worst-group accuracy from 81.26 to 89.04 on ResNet-50 and from 89.55 to 94.81 on DINOv2 ViT-B/14. On UrbanCars, it improves all three official shortcut gaps, including the combined background-plus-co-object gap. On PACO, it shifts the mean audited part–part interaction from -0.382 to $+0.339$, while natural Top-1 changes by $+0.13$ percentage points on a held-out PACO-LVIS test manifest covering five object categories. Further experiments retain Waterbirds gains with masks derived from the classifier itself and show larger interaction increases for the selected PACO part pair than for other audited part pairs. Interaction Steering enables direct supervision of how visual factors jointly affect a model's score, supporting both shortcut suppression and useful synergy.

1 INTRODUCTION

Modern vision models are trained through outcome supervision: the predicted class or relation should match the label. However, this supervision does not specify which visual factors should produce the prediction. A model may classify a bird correctly because of the bird, because of the background, or because of a shortcut between the two. It may recognize a car from its surrounding scene, infer an object relation from category co-occurrence, or miss useful evidence that only appears when informative parts are considered jointly. These failures are not only failures of accuracy; they are failures of interaction structure.

This point echoes a classical observation from Gestalt psychology: the whole is not merely the sum of its parts, but depends on the relations that bind them (Koffka, 1935, p. 176). Modern deep models often violate this principle in systematic ways. They rely on shortcut dependencies between an object and its background, between an object and a co-occurring object, or they fail to exploit useful dependencies between object parts (Geirhos et al., 2020; Beery et al., 2018; Xiao et al., 2021; Sagawa et al., 2020; Li et al., 2023). In this paper, we ask whether these interactions can be made explicit, measurable, and controllable.

Standard training via *Empirical Risk Minimization* (ERM) (Vapnik, 1999) rewards any predictive rule that lowers loss on the observed distribution, regardless of whether that rule has the intended compositional form. The core problem is that we lack a way to explicitly specify, measure, and enforce which visual factors should interact in the model's score. Existing approaches to out-of-distribution (OOD) generalization address related failures through environments, groups, examples, or classifier heads. Invariance-based methods (Arjovsky et al., 2019; Krueger et al., 2021; Rame et al., 2022) seek representations that are stable across environments, but often require explicit group structure or impose

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

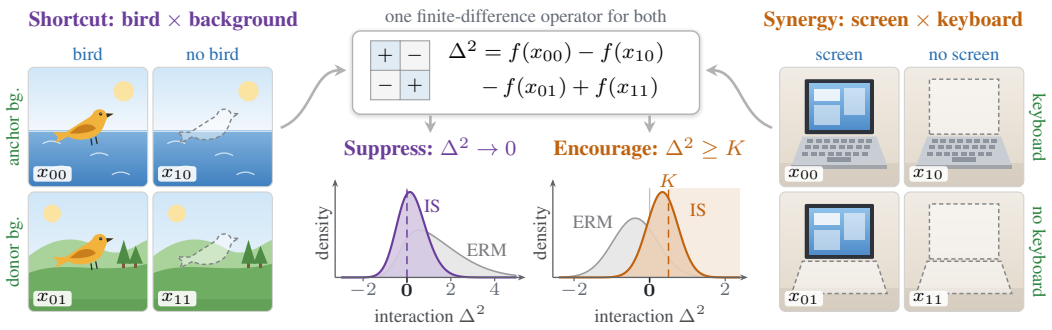


Figure 1: **Interaction Steering.** A named pair of visual factors defines a four-corner counterfactual lattice. The resulting finite-difference interaction is measured on model scores and steered toward a target: suppressing harmful dependencies with $\Delta^2 \to 0$ or encouraging useful synergies toward the lower-bound target $\Delta^2 \geq K > 0$. Counterfactual edits are used only for auxiliary interaction and guard losses.

broad constraints. Other robustness methods (Nam et al., 2020; Liu et al., 2021; Kirichenko et al., 2023) can improve worst-group behavior through reweighting or retraining. These approaches are effective baselines, but they do not directly name which visual factors should interact in the score, which interactions should be suppressed, or which should be encouraged.

Interaction Steering. We introduce *Interaction Steering via Counterfactual Finite Differences*, a framework for mechanism supervision in vision models. Interaction Steering applies finite-difference operators to counterfactual edits of a model's score, isolating named interactions such as object–background, object–co-object, and part–part effects. The same operators measure these score interactions and supply auxiliary training targets for changing them. Our main experiments study first- and second-order interactions, with an additional third-order experiment on UrbanCars.

Our pipeline is summarized in Figure 1. The practitioner first names the visual factors whose interaction should be audited, such as an object and its background. Counterfactual edits then instantiate the four lattice corners, corresponding to the chosen states of the two factors. A finite-difference stencil converts the four model scores into an interaction measurement. This measurement is then used as an auxiliary training signal: shortcut interactions are pushed toward zero, while beneficial synergies are pushed toward a positive target.

The key idea is to supervise not only the endpoint prediction, but also how the score changes under semantic edits. In Waterbirds (Sagawa et al., 2020), for example, the task is to classify images as "landbirds" or "waterbirds," while the background is spuriously correlated with the label. For an anchor image, we form four counterfactual corners by removing the bird, replacing the background, or applying both edits. The resulting finite difference estimates whether bird evidence remains stable across backgrounds or whether background modulates the bird's contribution. Interaction Steering turns this measurement into an auxiliary loss whose target is near-additivity between object and background.

Unlike training on labeled counterfactual examples (Chang et al., 2021; Noohdani et al., 2024), we use edited images only in auxiliary interaction and guard terms. The task loss remains on natural images. This separates learning to classify from specifying how selected factors should jointly contribute to the score.

Empirical Validation. Across three benchmarks, the same pattern holds: behavior improves when the targeted interaction changes in the intended direction. On Waterbirds, suppressing the object×background interaction improves worst-group accuracy from 81.26 to 89.04 on ResNet-50 and from 89.55 to 94.81 on DINOv2 ViT-B/14 (Oquab et al., 2024). On UrbanCars, a benchmark designed around multiple simultaneous shortcuts (Li et al., 2023), joint steering improves the official background gap from -13.84 to -3.77, co-occurring-object gap from -10.77 to -4.83, and combined gap from -71.71 to -16.57; single-target ablations show the expected Whac-A-Mole tradeoff. On PACO, a part-annotated object benchmark (Ramanathan et al., 2023), positive part×part steering preserves natural object classification within 0.13 points on a held-out PACO-LVIS test man-

ifest while shifting the mean audited interaction from -0.382 to $+0.339$ and increasing full-corner dominance from 35.35% to 79.19%. A matched PACO ablation further shows that the target-pair loss increases positive interaction with the guards and fine-tuning budget held fixed (Appendix I.7). Further experiments retain substantial Waterbirds gains with self-localized masks, directly suppress third-order UrbanCars interactions, and establish PACO target-pair specificity through paired uncertainty estimates (Appendices E.1, I.6, and I.7).

Contributions. Our contributions are the following:

1. We introduce Interaction Steering, a finite-difference framework for auditing named visual score interactions over counterfactual edits.

2. We define auxiliary steering and guard losses that suppress harmful interactions or encourage beneficial synergies, while keeping the primary task loss entirely on natural images and using counterfactual edits only for auxiliary losses.

3. We show empirically that named score interactions can be moved in the intended direction across shortcut-suppression regimes on Waterbirds and UrbanCars and part-synergy regimes on PACO, across multiple backbone families.

2 Related Work

Shortcut-learning and Background-bias. Models often rely on unreliable context, backgrounds, or other non-causal cues (Geirhos et al., 2020; Beery et al., 2018; Xiao et al., 2021; Sagawa et al., 2020) with benchmarks like Spawrious highlighting specific class–background correlations (Lynch et al., 2023). Existing mitigation strategies typically target groups or individual examples: GroupDRO optimizes for worst-group performance when labels are known (Sagawa et al., 2020), while LfF, JTT, and DFR reduce shortcut reliance via bias-amplified reweighting, high-loss upsampling, or balanced last-layer retraining (Nam et al., 2020; Liu et al., 2021; Kirichenko et al., 2023). These methods are strong baselines for Waterbirds-style failures, but they do not directly specify that an object's contribution should remain stable across backgrounds.

Multiple Shortcuts. Li et al. (2023) show that mitigating one shortcut can amplify another and introduce UrbanCars precisely to expose object–background and object–co-occurring-object failures. ExMap (Chakraborty et al., 2024) follows the same no-group-label setting, but still proceeds by inferring pseudo-groups from explanation heatmaps and then applying group-robust training. We directly target multiple shortcut interactions, rather than discovering latent groups and reweighting to suppress them.

Explanation-guided and Intervention-style Debiasing. Methods like Right for the Right Reasons and CDEP regularize gradients or decomposition-based explanations during training (Ross et al., 2017; Rieger et al., 2020), while MaskTune and Decompose-and-Compose suppress shortcuts by masking evidence or retraining on recomposed images (Asgari et al., 2022; Noohdani et al., 2024). Alternatively, interpretability tools like MAIA and self-reflective attribute methods discover or describe dependencies post hoc (Rott Shaham et al., 2024; Li et al., 2025). While we share this spirit, we differ in our controlled object: instead of penalizing saliency or training primarily on edited data, we use edited corners to compute auxiliary finite-difference and guard losses on specific interactions.

Counterfactual and Invariant Data-Generation. These methods apply prediction losses to individual edited images, often to break spurious associations or enforce invariance (Chang et al., 2021; Noohdani et al., 2024). Interaction Steering instead uses edited images to calculate finite-difference interaction among the model's scores at the corners, but does not apply the supervised task loss to edited corners.

Interaction Attribution. Shapley–Taylor attributes predictions to interactions up to a chosen order (Sundararajan et al., 2020), while Integrated Hessians quantifies pairwise feature interactions (Janizek et al., 2021). We use finite-difference interaction measurements as training objectives for suppressing harmful visual dependencies and encouraging useful synergy.

3 INTERACTION STEERING (IS)

Interaction Steering has three components: a finite-difference operator that quantifies interactions via counterfactual edits, an interaction loss that targets specific compositional rules, and task-specific instantiations that apply these tools to vision tasks. Together, these elements turn a named interaction into something that can be measured and trained directly. Figure 1 summarizes the overall workflow.

3.1 MEASURING INTERACTIONS WITH FINITE DIFFERENCES

A first step toward controlling how a model combines visual factors is to measure their interaction in a principled way. We introduce the first- and second-order operators below. In continuous settings, the interaction between two variables is captured by the mixed partial derivative $\frac{\partial^2 f}{\partial A \partial B}\big|_x$. Here, we construct its discrete analogue, applied not to infinitesimal coordinates but to a small counterfactual lattice.

1. **First-Order Effect (Δ^1).** Let $f(\cdot)$ denote the model, mapping an input image x to an output logit (or margin). Let $T_A(x)$ be a transformation that applies an intervention associated with visual factor A (such as removing an object or altering a color). The first-order effect of factor A is defined as the change in the model's output after the transformation:
$$\Delta^1_A(x) = f(T_A(x)) - f(x). \tag{1}$$
This quantity captures the main effect of A while holding all other factors fixed.
2. **Second-Order Interaction (Δ^2).** For two factors A and B, we construct four corners $x_{00}, x_{10}, x_{01}, x_{11}$ indexed by the chosen low/high semantic states of each axis. The indices name positions in the counterfactual lattice rather than universal presence labels, and each benchmark assigns concrete semantics to those positions. In the object–context convention used in Figure 1, x_{00} is the natural anchor, the first bit toggles object removal, and the second bit toggles the context change; thus x_{10} removes the object on the anchor, x_{01} places the object in an alternate context, and x_{11} applies both edits. The second-order finite difference is:
$$\Delta^2_{A,B}(x) = \big(f(x_{00}) - f(x_{10})\big) - \big(f(x_{01}) - f(x_{11})\big), \tag{2}$$
which is the discrete counterpart of the mixed partial derivative $\frac{\partial^2 f}{\partial A\, \partial B}$. A nonzero Δ^2 indicates that the effect of one intervention depends on whether the other is applied, while a value near zero indicates an *additive* relationship. In practice, the sign is interpreted with respect to the chosen intervention directions.

A selected k-factor interaction uses 2^k corners, where k counts the semantic factors in that interaction. Our experiments use $k \leq 3$, including direct third-order steering on UrbanCars (Appendix I.6). For larger k, sparse Möbius recovery (Kang et al., 2024) offers a possible extension for estimating interactions from fewer model evaluations (in polynomial time) under suitable sparsity assumptions; its use for steering remains future work (Appendix J).

3.2 THE INTERACTION STEERING LOSS

The finite-difference operator provides a quantitative measure of how two factors interact, but it also offers a natural lever for controlling this interaction during training. We convert this diagnostic quantity into a training objective through the *Interaction Steering Loss*. Given a natural training example x_{nat} and its associated counterfactual lattice, we write $\Delta^2_{A,B}(x_{\text{nat}})$ for the finite difference computed on that lattice. We instantiate f as the true-class margin and optimize the raw margin-space interaction
$$\mathcal{L}_\Delta(\mathcal{C}(x_{\text{nat}})) = \ell_{\text{steer}}\big(\Delta^2_{A,B}(x_{\text{nat}}); K, \tau\big), \tag{3}$$
where ℓ_{steer} is chosen according to the desired interaction behavior. In suppression settings, we use a squared dead-zone penalty
$$\ell_{\text{sup}}(\Delta; \tau) = [|\Delta| - \tau]^2_+. \tag{4}$$
For positive-synergy settings, we use either a lower-bound target
$$\ell_{\text{lb}}(\Delta; K) = [K - \Delta]^2_+ \tag{5}$$

or, when an upper cap is also desired, a band target

$$\ell_{\text{band}}(\Delta; K_\ell, K_u) = [K_\ell - \Delta]_+^2 + [\Delta - K_u]_+^2. \tag{6}$$

The interaction loss is added alongside the task loss on natural images,

$$\mathcal{L} = \mathcal{L}_{\text{task}}(x_{\text{nat}}, y) + \lambda_\Delta \, \mathcal{L}_\Delta(\mathcal{C}(x_{\text{nat}})) + \sum_j \gamma_j \, \mathcal{L}_{\text{guard},j}(\mathcal{C}(x_{\text{nat}})) \tag{7}$$

and plays a complementary role. Here $\mathcal{C}(x_{\text{nat}})$ denotes the counterfactual lattice associated with the natural example. The task term trains the model on unedited, natural data, while the interaction loss shapes how the model combines visual factors internally. The edited corners are used only inside auxiliary steering and guard terms and are never passed to $\mathcal{L}_{\text{task}}$. When one corner coincides with the natural image itself, that natural corner still carries the task loss. This separation keeps primary supervision on natural data, while the auxiliary terms adjust the mixed-partial structure of the learned function. The audit measures the raw margin-space interaction Δ^2 (or $|\Delta^2|$), which the steering losses also control. A model could satisfy the interaction constraint in degenerate ways, such as by collapsing the first-order effect of a factor so that Δ^1 becomes negligible, or by increasing synergy for the wrong class. In positive-synergy removal settings, a common first-order guard is an order-preservation penalty that keeps the natural corner above the edited corners. Writing m_{nat} for the natural corner score and m_A, m_B, m_{AB} for the three edited corners, one convenient form is

$$\ell_{\text{order}} = [q - (m_{\text{nat}} - m_A)]_+^2 + [q - (m_{\text{nat}} - m_B)]_+^2 + [q - (m_{\text{nat}} - m_{AB})]_+^2, \tag{8}$$

Here q is the desired score margin. Under the anchor-toggle convention, $m_{\text{nat}} = m_{00}$, $m_A = m_{10}$, $m_B = m_{01}$, and $m_{AB} = m_{11}$. For context-changing edits, guards can instead preserve the object's contribution while limiting changes caused by irrelevant context. Additional guards can penalize high positive interactions for non-target classes or squared logit drift from the initial model. Appendix D.1 details the loss instantiations.

Controlling Compositional Behavior. The suppression loss encourages near-additivity between factors, while a positive lower-bound target encourages super-additive interaction under the chosen intervention directions.

3.3 Instantiations for Vision Tasks

We consider two target behaviors: suppressing harmful object–context interactions and encouraging part–part synergy. The practitioner specifies the factors, target behavior, and edit procedure; corners are then generated automatically (Appendix B).

Suppressing Object–Context Interactions. To promote object-centric reasoning, we posit that an object's contribution to the class margin should not change merely because an irrelevant contextual cue changes. This does not require context to be useless; it targets cases where context modulates the object's evidence even though the context is not semantically required for the label. We therefore define $\Delta^2_{\text{obj}\times\text{ctx}}$ with corners x_{00}: object present under reference context c; x_{10}: object absent under c; x_{01}: object present under alternative context c'; and x_{11}: object absent under c'. Under this construction, Δ^2 is the change in the object's score contribution when the context changes. Suppressing Δ^2 is therefore a targeted local objective for removing context-dependent modulation of object evidence, while first-order guard losses keep the object itself predictive. The context axis can represent the background or a co-occurring object, with the remaining factors held fixed.

Encouraging Part–Part Object Synergy. To assess whether a model uses object-part evidence compositionally, we examine interactions between two informative part groups inside the same object crop. Our target is for the selected parts to provide super-additive evidence for the correct class. We therefore define $\Delta^2_{\text{partA}\times\text{partB}}$ with the same anchor-toggle convention used elsewhere in the paper: x_{00} is the natural crop with both selected parts visible, x_{10} removes part group A while keeping part group B, x_{01} removes part group B while keeping part group A, and x_{11} removes both selected part groups. For a true-class margin m, the audit quantity is $\Delta^2 = m_{11} - m_{10} - m_{01} + m_{00}$. Positive values mean that the two selected parts provide super-additive evidence for the correct class. We therefore set a positive lower-bound target for the true-class interaction and guard against high positive interactions for wrong classes on the same corners.

4 EXPERIMENTS

We evaluate suppression and synergy using the same finite-difference framework and true-class margin. Waterbirds (Sagawa et al., 2020) tests background reliance, while UrbanCars (Li et al., 2023) tests competing background and co-occurring-object shortcuts. Evidence that strong recognition can arise from additive aggregation of local patch logits (Brendel & Bethge, 2019) motivates our complementary tests of joint part evidence, using PACO's part annotations (Ramanathan et al., 2023) for controlled removal. Full counterfactual construction, training schedules, and model-selection rules are in Appendices D–F; unless otherwise noted, Waterbirds, UrbanCars, and PACO results are mean±standard deviation over three seeds.

4.1 SUPPRESSING BACKGROUND SHORTCUTS

Background shortcuts arise when scene context predicts training labels but becomes unreliable at test time. Waterbirds (Sagawa et al., 2020) isolates this failure through controlled bird–background correlations in landbird/waterbird classification. We report worst-group accuracy (WGA), the minimum accuracy over the four bird×background groups, and average accuracy. The construction extends to datasets allowing object localization and label-preserving background edits; UrbanCars below tests the same object×background interaction target alongside a second contextual shortcut.

Target Interaction and Edits. The desired rule is that bird evidence should be stable across backgrounds, so we suppress the raw object×background interaction toward $\Delta^2 \approx 0$. We construct object-removal and object-transfer edits using CUB segmentation masks, mask-guided inpainting, and cleansed donor backgrounds. In the convention used throughout Figure 2, x_{00} is the original anchor image, x_{10} removes the bird on that same anchor background, x_{01} pastes the bird onto a cleansed donor background, and x_{11} is the corresponding donor background with the bird removed. Construction details are in Appendix E.

Baselines. Across both backbones, we rerun Focal Loss (Lin et al., 2017), CVaR (Levy et al., 2020), Spectral Decoupling (Pezeshki et al., 2021), and JTT (Liu et al., 2021) without training group labels or masks, and GroupDRO (Sagawa et al., 2020) with training group labels. IS uses bird localization but no training group labels; all select checkpoints by validation WGA. Full CF-Aug uses the same masks and generator to add labeled bird–background compositions to cross-entropy (Appendix I.5).

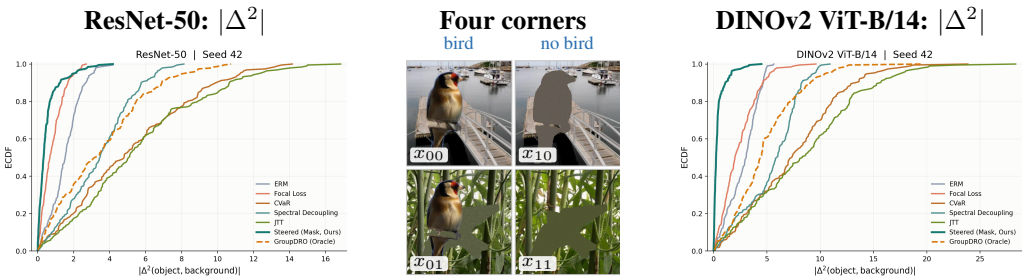
Results. Figure 2 shows that IS contracts raw $|\Delta^2|$ and improves WGA and average accuracy on both backbones. Ground-truth bird masks are not required for steering: on ResNet-50, a frozen COCO Mask R-CNN (He et al., 2017) segmenter yields $89.51 \pm 0.09\%$ WGA, while the ERM classifier's class-activation map (CAM) (Zhou et al., 2016) followed by GrabCut (Rother et al., 2004) yields $88.11 \pm 1.15\%$. Neither uses CUB masks or boxes for mask generation, ranking, or steering; CAM + GrabCut also uses no segmentation-pretrained model. Their interaction audits use released masks only at evaluation (Appendix E.1). Our ResNet-50 ExMap (Chakraborty et al., 2024) baseline reaches $86.67 \pm 1.92\%$ WGA versus $89.04 \pm 1.01\%$ for IS (Appendix I.4). On ResNet-50, full CF-Aug achieves higher WGA (92.52 vs. 89.04), while IS has smaller $|\Delta^2|$ (0.557 vs. 0.772) and lower background-class alignment on the birdless probe (58.7% vs. 66.8%). With validation-set DFR (Kirichenko et al., 2023), IS reaches 92.52% worst-group accuracy and 95.46% average test accuracy on ResNet-50 (Appendix I.3).

4.2 SUPPRESSING MULTIPLE CONTEXTUAL SHORTCUTS

Mitigating one contextual shortcut can amplify another (Li et al., 2023). UrbanCars was designed to expose this tradeoff through urban-versus-country car classification, with backgrounds and co-occurring objects that can be varied separately. Both context factors correlate with the car label during training; validation and test balance all 2^3 target×background×co-object groups. The joint-suppression construction extends to settings with separately editable contextual factors.

Target Interactions and Setup. We steer two pairwise interactions, object×background and object×co-occurring-object, using the same suppression loss (schedule in Appendix D.3).

Evaluation Metrics. ID accuracy weights groups by the training distribution. Each gap is rare-slice accuracy minus ID accuracy, so values closer to zero indicate better robustness. BG tests conflicting backgrounds with aligned co-objects; CoObj swaps these roles; BG+CoObj tests both conflicting.



Waterbirds test performance (3 seeds)

	ResNet-50		DINOv2 ViT-B/14	
Method	**WGA**	**Avg.**	**WGA**	**Avg.**
ERM	81.26±0.90	93.03±1.03	89.55±1.91	94.96±0.93
Focal Loss	79.94±1.85	91.65±1.36	88.29±3.03	95.48±0.50
CVaR	82.45±2.65	92.47±0.69	89.40±0.99	95.16±0.85
Spectral Decoupling	82.65±1.54	92.29±1.51	88.32±2.72	95.65±0.87
JTT	79.54±2.07	90.62±2.36	86.72±2.07	93.38±0.76
Interaction Steering (ours)	**89.04±1.01**	**95.54±0.51**	**94.81±0.48**	**97.72±0.31**
GroupDRO (group labels)	88.42±1.13	94.82±0.53	94.01±0.69	96.86±0.79

Figure 2: **Waterbirds.** Top: ECDFs of raw $|\Delta^2_{\text{obj}\times\text{bg}}|$ for ResNet-50 and DINOv2 ViT-B/14, with the four-corner construction shown at the same height. Bottom: 3-seed test performance.

Method	**BG**	**CoObj**	**Both**	**Method**	**BG**	**CoObj**	**Both**
ERM (paper)	−15.30	−11.20	−69.20	EIIL (E=1)	−4.20	−24.70	−44.90
Mixup	−12.60	−9.30	−61.80	JTT (E=2)	−23.30	−5.30	−52.10
CutMix	−45.00	−4.80	−86.50	EIIL (E=2)	−21.50	−6.80	−49.60
Cutout	−15.80	−10.40	−71.40	DebiAN	−14.90	−10.50	−69.00
AugMix	−10.30	−12.10	−70.20	LLE	−2.10	−2.70	−5.90
SD	−15.00	−3.60	−36.10	ERM (ours)	−13.84	−10.77	−71.71
CF+F Aug	−16.00	0.40	−19.40	BG-only (ours)	−4.29 ↑	−13.62 ↓	−37.75 ↑
LfF	−11.60	−18.40	−63.20	Co-only (ours)	−18.78 ↓	−4.25 ↑	−50.51 ↑
JTT (E=1)	−8.10	−13.30	−40.10	**IS (ours)**	**−3.77** ↑	**−4.83** ↑	**−16.57** ↑

Both = BG + CoObj. Mean gaps (pp); closer to 0 is better.
↑ / ↓ : improved / worse mean gap vs. our ERM.
Per metric: 1st 2nd 3rd

Figure 3: **UrbanCars.** All 14 published baseline configurations (Li et al., 2023) and four matched runs. Shading marks the three smallest mean gap magnitudes per metric. Arrows expose opposing single-target regressions; joint IS improves all three gaps. ID accuracy and three-seed standard deviations are in Table 15.

Figure 3 compares all reported methods on all three gaps; Appendix I.6 gives all four metrics and both pairwise interaction audits.

Results. Joint steering improves all three gaps: BG from -13.84 to -3.77, CoObj from -10.77 to -4.83, and BG+CoObj from -71.71 to -16.57, with ID accuracy decreasing from 97.84 to 95.10. The single-target ablations expose Whac-A-Mole behavior (Li et al., 2023): BG-only improves BG but worsens CoObj, while Co-only has the opposite tradeoff. Joint steering improves both, and both targeted interaction distributions contract (Appendix Figure 5).

Joint IS ranks second on the combined gap, behind LLE and ahead of CF+F Aug. LLE (Li et al., 2023) uses three last-layer heads on a shared backbone: one for original images and two for mask-based BG/CoObj augmentations, without shortcut labels. A learned shift classifier weights their logits per test image. This specialization and adaptive ensembling may help on the doubly conflicting slice; IS steers both interactions in a single classifier. Adding the third-order interaction to the target reduces mean $|\Delta^3|$ from 1.092 to 0.970 across three seeds on the validation audit, with a combined gap of -15.33 (ID 95.06%; Appendix I.6).

Table 1: **PACO held-out test behavior with a masking baseline.** All models are ResNet-50 object classifiers evaluated on the held-out PACO-LVIS test manifest.

Method	Natural Top-1	One-part Top-1	Both-masked Top-1	Sel. part drop	Mean Δ^2	$\Delta^2 > K$	Full-corner dom.	Wrong Δ^2
ResNet-50 ERM	97.66±0.38	93.99±1.53	85.02±4.06	0.359±0.105	−0.382±0.041	6.77±1.95	35.35±2.61	0.083±0.030
PartDrop	97.22±0.34	**96.97±0.36**	**95.88±0.56**	0.026±0.024	−0.055±0.026	4.46±0.57	22.54±3.48	0.036±0.016
Interaction Steering	**97.79±0.12**	96.07±0.32	93.81±0.42	**0.811±0.045**	**0.339±0.029**	**34.74±1.19**	**79.19±2.43**	**−0.139±0.008**

4.3 Encouraging Part–Part Synergy

Part–part synergy asks whether selected parts provide super-additive class evidence. PACO (Ramanathan et al., 2023) enables controlled removal through object and part annotations; the construction extends to other categories with credible part-removal edits. We train vision-only ResNet-50 classifiers on natural crops from five categories: bicycle, bottle, car, chair, and laptop. Checkpoints maximize natural validation Top-1; evaluation uses 2,516 held-out PACO-LVIS test objects.

Target Interaction and Edits. For each category, we define one semantically meaningful part pair and construct four anchor-toggle corners on the object crop: x_{00} is the natural crop with both selected parts visible, x_{10} removes part group A while keeping part group B, x_{01} removes part group B while keeping part group A, and x_{11} removes both selected part groups. All other object pixels remain visible. For the true-class margin m, $\Delta^2 = m_{11} - m_{10} - m_{01} + m_{00}$, where positive values mean that the two selected parts provide super-additive evidence for the correct object class. Figure 4 shows one audited corner stencil. Steering uses the lower-bound target of Section 3 with $K = 0.5$ and the guards described there; Appendix E.3 gives the corner-construction details.

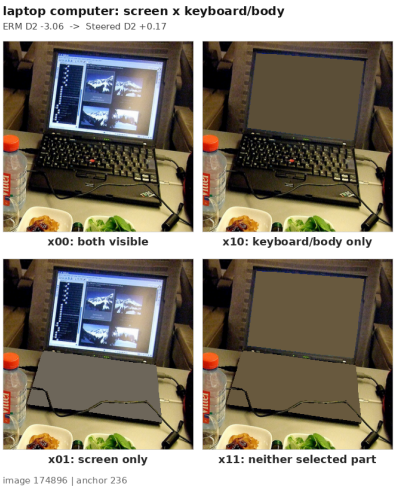


Figure 4: **PACO part-removal stencil.** One laptop screen×keyboard/body.

Baselines and Metrics. Besides ERM, we include a diagnostic *PartDrop* baseline. PartDrop trains with ordinary cross-entropy on randomly sampled PACO corners: the natural crop x_{00} with probability 0.5, and one of x_{10}, x_{01}, or x_{11} with total probability 0.5. This baseline tests whether generic exposure to masked part images is enough to reproduce the targeted part-pair mechanism.

We report natural Top-1 on x_{00}, one-part-masked Top-1 averaged over x_{10} and x_{01}, both-parts-masked Top-1 on x_{11}, the selected-part margin drop $\frac{1}{2}[(m_{00} - m_{01}) + (m_{00} - m_{10})]$, mean Δ^2, the fraction of anchors above $K = 0.5$, full-corner dominance, and the top wrong-class interaction. Full-corner dominance is the fraction of anchors for which $m_{00} > \max(m_{10}, m_{01}, m_{11})$.

Held-Out Test Results. Relative to ERM, IS changes natural Top-1 by $+0.13$ points while shifting mean target-pair Δ^2 from -0.382 to $+0.339$ (Table 1; ECDF in Appendix Figure 6). The fraction above $K = 0.5$ rises from 6.77% to 34.74%, and full-corner dominance from 35.35% to 79.19%. Selected-part margin drops increase, while the top wrong-class interaction decreases from 0.083 to -0.139. PartDrop improves masked-corner accuracy but reduces the selected-part drop to 0.026 and leaves the target interaction near zero.

With the same fine-tuning budget and guards, removing the target-pair loss leaves mean Δ^2 at -0.102, versus $+0.339$ for full IS (paired gain 0.441; 95% CI $[0.408, 0.479]$). Natural Top-1 is 97.99% versus 97.79%, respectively, while full-corner dominance is 78.68% versus 79.19% (Appendix I.7).

Specificity to the Steered Pair. We also audit part pairs that were not optimized. For each held-out test instance, we compare the target pair, two mixed pairs that share one steered part, and a fully off-target pair made from two non-steered parts. Table 2 reports the mean interaction for each pair.

The target-pair interaction gain is $+0.720$ (95% CI $[0.647, 0.790]$). On common anchors, it exceeds the larger mixed-pair change by 0.356 (CI $[0.301, 0.419]$) and the off-target change by 0.454 (CI $[0.355, 0.576]$). Fully off-target pairs remain near zero (Table 2). These paired contrasts support

Table 2: **PACO off-target pair audit on held-out test anchors.** The target pair is the optimized semantic pair; mixed pairs share one steered part; fully off-target pairs contain two non-steered parts.

Audited pair	Rows	ERM mean Δ^2	PartDrop mean Δ^2	IS mean Δ^2	Change (IS − ERM)
Target steered pair	2516	-0.382 ± 0.041	-0.055 ± 0.026	$\mathbf{0.339 \pm 0.029}$	$\mathbf{+0.720 \pm 0.065}$
Mixed-A pair	1915	-0.172 ± 0.063	-0.035 ± 0.010	0.027 ± 0.020	$+0.200 \pm 0.044$
Mixed-B pair	1915	-0.172 ± 0.066	-0.031 ± 0.014	0.121 ± 0.040	$+0.293 \pm 0.038$
Fully off-pair	1070	-0.037 ± 0.016	-0.032 ± 0.019	0.031 ± 0.015	$+0.068 \pm 0.003$

specificity to the named pair; Appendix I.7 describes the crossed-cluster bootstrap and eligible anchor sets.

PACO therefore provides evidence that the same finite-difference objective can encourage beneficial visual synergy, not only suppress harmful shortcut interactions. With checkpoints fixed, interaction gains persist across four part-removal fills on the validation manifest (Table 3; full diagnostics in Appendix I.7). Appendix I.8 reports a PSG relation-synergy diagnostic under strong category-pair priors.

Table 3: **PACO robustness to part-removal fills.** IS-minus-ERM changes on the validation manifest with fixed checkpoints. Percentage changes are in points.

Fill	Mean Δ^2	$\Delta^2 > 0.5$ (%)	Full-corner dom. (%)	Wrong Δ^2
Mean background	$+0.648$	$+25.01$	$+42.99$	-0.210
Blur	$+0.361$	$+13.49$	$+49.12$	-0.113
Black	$+0.635$	$+30.03$	$+43.42$	-0.238
Mixed	$+0.739$	$+24.39$	$+45.77$	-0.235

5 Limitations and Broader Impact

Limitations. IS requires named factors and credible edits. Poor masks, inpainting or donor artifacts, and ambiguous or heavily occluded factors can invalidate the corners. Predicted-mask and fill-mode diagnostics test edit variation; sensitivity to other steering targets and guard weights remains uncharacterized. New tasks require suitable generators, guards, and validation criteria (Appendix B). Evaluating 2^k corners per selected interaction adds training cost despite offline preparation; inference uses only the classifier (Appendix G). UrbanCars still trails LLE on the hardest slice. PACO evaluates controlled removals in five annotated categories, not natural occlusion or unseen categories.

Broader Impact. Reporting the targeted dependency and its measured change makes robustness interventions more transparent. Chosen targets and factor annotations can encode harmful biases, while unaudited interactions remain unaddressed. Reports should specify annotation sources, counterfactual transformations, and residual failures. We use public benchmarks and pretrained backbones and make no safety-critical deployment claims.

6 Conclusion

Robustness failures are often best understood as problematic interactions where a model's score for one factor depends improperly on another, such as birds changing with backgrounds or shortcuts persisting despite partial mitigation. Interaction Steering makes these dependencies actionable by organizing counterfactual edits into a semantic lattice; this transforms visual dependencies into finite-difference targets that can both audit behavior and serve as auxiliary training objectives. Across various settings, targeting these interactions moves model behavior in intended directions—contracting object-context dependencies or increasing part-pair synergy. This allows us to move beyond simple performance metrics to specify and evaluate the precise visual mechanisms that make models successful.

Reproducibility Statement

Dataset splits and seeds are documented in Appendices A and C; losses, training schedules, and counterfactual construction in Appendices D–E; and model selection and statistical reporting in Appendix F. Appendix G gives approximate compute budgets, and Appendix H identifies public datasets and pretrained assets. Code is not released.

AI Use Statement

We used generative AI tools, including OpenAI Codex and Anthropic Claude Code, to assist with manuscript editing, literature search, figure creation and coding. The authors reviewed and revised the AI-assisted material and take responsibility for the final content, claims, and reported results.

References

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019. doi: 10.48550/arXiv.1907.02893. URL https://arxiv.org/abs/1907.02893.

Saeid Asgari, Aliasghar Khani, Fereshte Khani, Ali Gholami, Linh Tran, Ali Mahdavi Amiri, and Ghassan Hamarneh. MaskTune: Mitigating spurious correlations by forcing to explore. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/93be245fce00a9bb2333c17ceae4b732-Abstract-Conference.html.

Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European Conference on Computer Vision*, pp. 456–473, 2018. URL https://openaccess.thecvf.com/content_ECCV_2018/html/Beery_Recognition_in_Terra_ECCV_2018_paper.html.

Wieland Brendel and Matthias Bethge. Approximating CNNs with bag-of-local-features models works surprisingly well on ImageNet. In *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1904.00760.

Rwiddhi Chakraborty, Adrian Sletten, and Michael C. Kampffmeyer. ExMap: Leveraging explainability heatmaps for unsupervised group robustness to spurious correlations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12017–12026, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Chakraborty_ExMap_Leveraging_Explainability_Heatmaps_for_Unsupervised_Group_Robustness_to_Spurious_CVPR_2024_paper.html.

Chun-Hao Chang, George Alexandru Adam, and Anna Goldenberg. Towards robust classification model by counterfactual and invariant data generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15212–15221, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Chang_Towards_Robust_Classification_Model_by_Counterfactual_and_Invariant_Data_Generation_CVPR_2021_paper.html.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009. doi: 10.1109/CVPR.2009.5206848. URL https://www.image-net.org/static_files/papers/imagenet_cvpr09.pdf.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020. doi: 10.1038/s42256-020-00257-z. URL https://doi.org/10.1038/s42256-020-00257-z.

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

(CVPR), pp. 770–778, 2016. URL https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2961–2969, 2017. URL https://openaccess.thecvf.com/content_iccv_2017/html/He_Mask_R-CNN_ICCV_2017_paper.html.

Joseph D. Janizek, Pascal Sturmfels, and Su-In Lee. Explaining explanations: Axiomatic feature interactions for deep networks. *Journal of Machine Learning Research*, 22(104):1–54, 2021. URL https://jmlr.org/papers/v22/20-1223.html.

Justin Singh Kang, Yigit Efe Erginbas, Landon Butler, Ramtin Pedarsani, and Kannan Ramchandran. Learning to understand: Identifying interactions via the Möbius transform. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. Last layer re-training is sufficient for robustness to spurious correlations. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Zb6c8A-Fghk.

Kurt Koffka. *Principles of Gestalt Psychology*. Harcourt, Brace and Company, New York, 1935.

David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (REx). In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5815–5826. PMLR, 2021. URL https://proceedings.mlr.press/v139/krueger21a.html.

Daniel Levy, Yair Carmon, John C. Duchi, and Aaron Sidford. Large-scale methods for distributionally robust optimization. In *Advances in Neural Information Processing Systems*, volume 33, pp. 8847–8860, 2020. URL https://papers.nips.cc/paper/2020/hash/64986d86a17424eeac96b08a6d519059-Abstract.html.

Christy Li, Josep Lopez Camuñas, Jake Thomas Touchet, Jacob Andreas, Agata Lapedriza, Antonio Torralba, and Tamar Rott Shaham. Automated detection of visual attribute reliance with a self-reflective agent. In *Advances in Neural Information Processing Systems*, 2025. URL https://nips.cc/virtual/2025/poster/118510.

Zhiheng Li, Ivan Evtimov, Albert Gordo, Caner Hazirbas, Tal Hassner, Cristian Canton Ferrer, Chenliang Xu, and Mark Ibrahim. A Whac-a-Mole dilemma: Shortcuts come in multiples where mitigating one amplifies others. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20071–20082, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Li_A_Whac-a-Mole_Dilemma_Shortcuts_Come_in_Multiples_Where_Mitigating_One_CVPR_2023_paper.html.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Computer Vision – ECCV 2014*, volume 8693 of *Lecture Notes in Computer Science*, pp. 740–755. Springer, 2014. doi: 10.1007/978-3-319-10602-1_48. URL https://doi.org/10.1007/978-3-319-10602-1_48.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2980–2988, 2017. doi: 10.1109/ICCV.2017.324. URL https://openaccess.thecvf.com/content_iccv_2017/html/Lin_Focal_Loss_for_ICCV_2017_paper.html.

Evan Z. Liu, Behzad Haghgoo, Annie S. Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. Just train twice: Improving group robustness without training group information. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 6781–6792. PMLR, 2021. URL https://proceedings.mlr.press/v139/liu21f.html.

Aengus Lynch, Gbètondji J.-S. Dovonon, Jean Kaddour, and Ricardo Silva. Spawrious: A benchmark for fine control of spurious correlation biases. *arXiv preprint arXiv:2303.05470*, 2023. doi: 10.48550/arXiv.2303.05470. URL https://arxiv.org/abs/2303.05470.

Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: De-biasing classifier from biased classifier. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/hash/eddc3427c5d77843c2253f1e799fe933-Abstract.html.

Fahimeh Hosseini Noohdani, Parsa Hosseini, Aryan Yazdan Parast, Hamidreza Yaghoubi Araghi, and Mahdieh Soleymani Baghshah. Decompose-and-Compose: A compositional approach to mitigating spurious correlation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27662–27671, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Noohdani_Decompose-and-Compose_A_Compositional_Approach_to_Mitigating_Spurious_Correlation_CVPR_2024_paper.html.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=a68SUt6zFt.

Mohammad Pezeshki, Sékou-Oumar Kaba, Yoshua Bengio, Aaron C. Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. In *Advances in Neural Information Processing Systems*, volume 34, pp. 1256–1272, 2021. URL https://papers.nips.cc/paper/2021/hash/0987b8b338d6c90bbedd8631bc499221-Abstract.html.

Vignesh Ramanathan, Anmol Kalia, Vladan Petrovic, Yi Wen, Baixue Zheng, Baishan Guo, Rui Wang, Aaron Marquez, Rama Kovvuri, Abhishek Kadian, Amir Mousavi, Yiwen Song, Abhimanyu Dubey, and Dhruv Mahajan. PACO: Parts and attributes of common objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7141–7151, 2023. doi: 10.1109/CVPR52729.2023.00690. URL https://openaccess.thecvf.com/content/CVPR2023/html/Ramanathan_PACO_Parts_and_Attributes_of_Common_Objects_CVPR_2023_paper.html.

Alexandre Rame, Corentin Dancette, and Matthieu Cord. Fishr: Invariant gradient variances for out-of-distribution generalization. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 18347–18377. PMLR, 2022. URL https://proceedings.mlr.press/v162/rame22a.html.

Laura Rieger, Chandan Singh, William Murdoch, and Bin Yu. Interpretations are useful: Penalizing explanations to align neural networks with prior knowledge. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8116–8126. PMLR, 2020. URL https://proceedings.mlr.press/v119/rieger20a.html.

Andrew Slavin Ross, Michael C. Hughes, and Finale Doshi-Velez. Right for the right reasons: Training differentiable models by constraining their explanations. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, pp. 2662–2670, 2017. doi: 10.24963/ijcai.2017/371. URL https://www.ijcai.org/Proceedings/2017/371.

Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. GrabCut: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23(3):309–314, 2004. doi: 10.1145/1015706.1015720. URL https://doi.org/10.1145/1015706.1015720.

Tamar Rott Shaham, Sarah Schwettmann, Franklin Wang, Achyuta Rajaram, Evan Hernandez, Jacob Andreas, and Antonio Torralba. A multimodal automated interpretability agent. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 44293–44321. PMLR, 2024. URL https://proceedings.mlr.press/v235/shaham24a.html.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=ryxGuJrFvS.

Mukund Sundararajan, Kedar Dhamdhere, and Ashish Agarwal. The Shapley-Taylor interaction index. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9259–9268. PMLR, 2020. URL https://proceedings.mlr.press/v119/sundararajan20a.html.

Vladimir N. Vapnik. *The Nature of Statistical Learning Theory*. Information Science and Statistics. Springer, New York, 1999. doi: 10.1007/978-1-4757-3264-1. URL https://link.springer.com/book/10.1007/978-1-4757-3264-1.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-UCSD birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. URL https://authors.library.caltech.edu/records/cvm3y-5hh21.

Kai Xiao, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. Noise or signal: The role of image backgrounds in object recognition. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=gl3D-xY7wLq.

Jingkang Yang, Yi Zhe Ang, Zujin Guo, Kaiyang Zhou, Wayne Zhang, and Ziwei Liu. Panoptic scene graph generation. In *Computer Vision – ECCV 2022*, volume 13687 of *Lecture Notes in Computer Science*, pp. 178–196. Springer, 2022. doi: 10.1007/978-3-031-19812-0_11. URL https://www.ecva.net/papers/eccv_2022/papers_ECCV/html/222_ECCV_2022_paper.php.

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2921–2929, 2016. doi: 10.1109/CVPR.2016.319. URL https://openaccess.thecvf.com/content_cvpr_2016/html/Zhou_Learning_Deep_Features_CVPR_2016_paper.html.

Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6):1452–1464, 2018. doi: 10.1109/TPAMI.2017.2723009. URL https://doi.org/10.1109/TPAMI.2017.2723009.

A Experimental Conventions

For IS, counterfactual images enter auxiliary steering and guard losses; the primary supervised loss uses natural images. Full CF-Aug and PartDrop instead apply cross-entropy to their eligible edited examples.

For the main quantitative tables, the reported seed sets are fixed: Waterbirds uses three seeds per backbone $(1, 21, 42)$, UrbanCars uses three seeds $(0, 1, 2)$, and PACO uses three seeds $(0, 1, 2)$. The PSG appendix diagnostic uses three seeds $(0, 1, 3)$.

Unless otherwise noted, quantitative summaries report the mean and sample standard deviation across these fixed runs, not the standard error. Single-run diagnostics are labeled separately.

B Benchmark-Specific Instantiations

To adapt IS, name the factors and desired relationship, define the corners, apply the finite-difference target to the task score, and choose guards for behavior to preserve. Factor choices are task-level, including five category-level part pairs in PACO; no image is hand-edited. Generation, filtering, and sampling are automatic, while generators, guards, and selection rules remain task-specific. Table 4 lists inputs; Appendices D.1, E, and F specify scores, construction, and selection.

Table 4: **Benchmark inputs and interaction targets.** All settings use natural training images and labels; listed factor information supports programmatic corner generation. Full protocols and selection rules are in Appendices C–F.

Setting	Required factor information	Corner construction	Score / target
Waterbirds	Bird localization: CUB masks, frozen COCO Mask R-CNN, or ERM CAM + GrabCut.	Remove anchor and donor birds; paste the anchor bird onto the cleaned donor background.	True-class margin; object×background suppression.
UrbanCars	Object/co-object masks, benchmark factor variants, and alignment buckets for bank construction.	Read the object/background/co-object lattice; select pairwise or three-way corners.	Class margin; stagewise shortcut suppression.
PACO	Object/part annotations; one specified part pair per category (five pairs).	Remove either or both selected parts within the object crop using masking fills.	True-class margin; positive part-pair target.
PSG	Subject/object panoptic masks and predicate annotations.	Remove the subject, object, or both within the relation crop.	True-predicate score; lower/band target.

C Datasets, Splits, and Evaluation Protocols

C.1 Waterbirds

Waterbirds follows the standard benchmark introduced in Sagawa et al. (2020). The task is binary classification between landbirds and waterbirds, with a spuriously correlated background. We use the official train/validation/test split and evaluate both average accuracy and worst-group accuracy (WGA), where the worst-group score is the minimum accuracy across the four bird×background groups. The Waterbirds tables rerun a suite of strong baselines that do not use group labels during training under the matched backbone-specific setup and report mean±std over three seeds per backbone. Our steered Waterbirds method is evaluated on the same splits and backbones and also does not use group labels during training, with donor backgrounds sampled randomly.

C.2 UrbanCars

UrbanCars is taken from the Whac-A-Mole benchmark Li et al. (2023). Each example is defined by three binary factors: the target object label, the background label, and the co-occurring-object label, yielding $2^3 = 8$ groups. The training split uses shortcut correlation 0.95 for both background and co-occurring object, while validation and test are balanced across groups. We report the official benchmark metrics: in-distribution accuracy (ID Acc), BG Gap, CoObj Gap, and BG+CoObj Gap. For the matched ERM and steered rows in the main paper, all UrbanCars numbers are mean±std over three random seeds.

C.3 PACO

PACO provides object and part annotations for common object categories Ramanathan et al. (2023). We use the PACO-LVIS annotations with COCO 2017 images and construct a controlled object-crop classification task over five categories: bicycle, bottle, car, chair, and laptop. We form deterministic train/validation/test splits from the corresponding PACO-LVIS v1 annotation files by retaining object instances in the five target categories that contain the required target parts and satisfy fixed size filters: object area at least 0.002 of image area, each target part group at least 5×10^{-5} of image area, crop area at most 0.95 of image area, and at least 10 training examples per retained class. This yields 10,998 training object instances, 697 validation object instances for checkpoint selection, and 2,516 held-out test object instances for final PACO reporting. The splits are image-disjoint by COCO `image_id`, and the test split is not used for checkpoint selection. For each category, we choose one semantically meaningful part pair: bicycle wheel×handlebar/frame, bottle cap/neck×body, car wheel×window/body, chair seat×back, and laptop screen×keyboard/body. The main PACO results report mean±std over three seeds on the held-out PACO-LVIS test split and focus on controlled

part-removal behavior in addition to natural object classification. Because PACO-LVIS does not provide an official evaluation server for this constructed five-category task, we report final PACO results on this deterministic held-out test split.

C.4 PSG

PSG follows the panoptic scene graph dataset introduced by Yang et al. (2022), using COCO images Lin et al. (2014), OpenPSG-style relation annotations, and panoptic segmentation masks. We do not evaluate full scene-graph generation. Instead, each example is a ground-truth subject/object relation pair, and the model predicts a multi-label vector over 12 selected predicates from a crop around the subject and object masks. The manifest represents each pair with a multi-hot predicate target, and audit rows expand positive predicates into one target row per relation. We report natural predicate classification metrics on the selected predicate set together with mechanism metrics on audited relation anchors.

D Training Details and Hyperparameters

D.1 Common Loss Components

The benchmark-specific objectives instantiate a small shared family of steering and guard losses. For a scalar interaction value Δ, a true-class margin quartet $(m_{00}, m_{10}, m_{01}, m_{11})$, and non-target interaction values Δ^{wrong}, the common components are:

$$\ell_{\text{sup}}(\Delta;\tau) = [|\Delta| - \tau]_+^2, \tag{9}$$

$$\ell_{\text{lb}}(\Delta;K) = [K - \Delta]_+^2, \tag{10}$$

$$\ell_{\text{band}}(\Delta;K_\ell,K_u) = [K_\ell - \Delta]_+^2 + [\Delta - K_u]_+^2, \tag{11}$$

$$\ell_{\text{order}} = [q - (m_{\text{nat}} - m_A)]_+^2 + [q - (m_{\text{nat}} - m_B)]_+^2 + [q - (m_{\text{nat}} - m_{AB})]_+^2, \tag{12}$$

$$\ell_{\text{wrong}} = \frac{1}{k} \sum_{c \in \text{TopKWrong}} [\Delta_c^{\text{wrong}} - \tau_{\text{wrong}}]_+^2, \tag{13}$$

$$\ell_{\text{stab}} = \|z_\theta(x_{\text{nat}}) - z_{\text{ERM}}(x_{\text{nat}})\|_2^2. \tag{14}$$

Here $[\cdot]_+ = \max(\cdot, 0)$ and z_θ denotes the current model logits on natural images. UrbanCars additionally uses a reference-fraction hinge penalty for the targeted shortcut interactions,

$$\ell_{\text{ref}}(\Delta;\rho,\Delta_{\text{ref}}) = [|\Delta| - \rho|\Delta_{\text{ref}}|]_+^2, \tag{15}$$

where Δ_{ref} is the audited warm-start interaction magnitude for the same anchor.

The score m is benchmark-specific but explicit. For single-label logits $z \in \mathbb{R}^C$ and target class y, Waterbirds, UrbanCars, and PACO use the true-class margin

$$m_{\text{cls}}(z,y) = z_y - \max_{c \neq y} z_c. \tag{16}$$

For binary Waterbirds this reduces to the difference between the true-class logit and the other class logit. PSG is multi-label. In its warm stage, the `ovr_margin` score mode (*one-vs-rest margin*) uses

$$m_{\text{ovr}}(z,p;\mathcal{P}) = z_p - \log \sum_{q \notin \mathcal{P}} \exp(z_q), \tag{17}$$

where p is the audited target predicate and $\mathcal{P}$ is the set of positive predicates for that pair. In the final PSG stage, `true_logit` uses the raw target-predicate logit

$$m_{\text{logit}}(z,p) = z_p. \tag{18}$$

PSG also uses two extra guard terms in the reported recipe:

$$\ell_{\text{prior-neg}} = [\Delta^{\text{prior}} - \tau_{\text{prior}}]_+^2, \tag{19}$$

$$\ell_{\text{nat}} = [m_{\text{nat}}^{\text{ERM}} + \gamma_{\text{nat}} - m_{\text{nat}}]_+^2, \tag{20}$$

where Δ^{prior} is the interaction on the category-pair-prior-favored wrong predicate and $m_{\text{nat}}^{\text{ERM}}$ is the warm-start reference score on the natural anchor corner. Under the paper's anchor-toggle convention for PACO and PSG, $m_{\text{nat}} = m_{00}$, $m_A = m_{10}$, $m_B = m_{01}$, and $m_{AB} = m_{11}$. The reported PSG checkpoint selector `best_by_map_floor_synergy` keeps only checkpoints with validation mAP above a floor η and then maximizes

$$S_{\text{floor}} = \overline{\Delta^2} + 0.25\,\text{Frac}(\Delta^2 > 0.5) + 0.10\,\text{CornerTop} - 0.10\,\overline{\text{WrongGuard}}, \quad \text{s.t. mAP} \geq \eta. \tag{21}$$

The hyperparameter names in the benchmark recipes map directly to these structural terms. Waterbirds uses `lambda_d2` for the interaction weight and implements its guards through `guard_weight`, the object-effect floor terms, and the object-effect reward term. UrbanCars uses `lambda_bg` and `lambda_co` for the two optimized pairwise shortcut terms; the three-way term is unoptimized in the primary pairwise setting and is additionally targeted in the experiment of Appendix I.6 , with guard/reward coefficients such as `lam_obj_guard`, `lam_obj_ctx_guard`, `lam_bg_main`, and `lam_co_main`. PACO uses `lambda_d2`, `lambda_order`, `lambda_wrong`, and `lambda_stability` for ℓ_{lb}, ℓ_{order}, ℓ_{wrong}, and ℓ_{stab}, respectively. PSG uses the same order, wrong-predicate, and stability components together with `lambda_prior_neg` for $\ell_{\text{prior-neg}}$ and the parameter `lambda_anchor_x11` for the natural-corner guard ℓ_{nat}.

D.2 WATERBIRDS

The reported Waterbirds reruns use AdamW with cosine decay, label smoothing 0.1, AMP, and a two-stage procedure consisting of an ERM warm start followed by steered fine-tuning. The non-group-label baselines use the same backbone-specific pipeline with their algorithm-specific objectives or schedules; we do not perform an exhaustive per-method hyperparameter search. For ResNet-50, ERM uses 30 epochs, batch size 128, learning rate 2×10^{-4}, and weight decay 5×10^{-4}. The steered stage then fine-tunes for 20 epochs with batch size 128, learning rate 10^{-5}, weight decay 10^{-4}, and two frozen-backbone epochs. The interaction loss uses $\lambda_\Delta = 0.05$, $\tau = 0.05$, a squared-hinge suppression penalty ℓ_{sup}, an audit-ranked anchor bank formed by taking the top 55% of anchors by interaction magnitude and balancing the retained set by class label, two donor backgrounds per anchor, and 48 anchors per update. The final reported setting uses `guard_weight`$= 0.13$, `lam_obj`$=$ 0.6, `lam_obj_floor`$= 0.3$, `tau_bg_min`$= 10.0$, `tau_bg_fac`$= 0.0$, `tau_bg_guard_min`$= 0.5$, `tau_bg_guard_fac`$= 0.02$, and `conf_min`$= 0.0$. Donors are sampled uniformly at random from the valid donor pool, excluding the anchor itself.

For DINOv2 ViT-B/14, ERM uses 30 epochs, batch size 128, learning rate 10^{-4}, weight decay 10^{-4}, five frozen-backbone epochs, and backbone learning-rate multiplier 0.1. The steered stage again runs for 20 epochs with learning rate 10^{-5}, weight decay 10^{-4}, $\lambda_\Delta = 0.05$, $\tau = 0.05$, and two frozen-backbone epochs, but uses batch size 64 and 24 anchors per step. For both Waterbirds backbones, the reported accuracy and ECDF audits use the same validation-WGA-selected checkpoint for each seed.

D.3 URBANCARS

Our matched UrbanCars runs use a ResNet-50 backbone , batch size 128, momentum SGD with momentum 0.9, weight decay 10^{-4}, and AMP. The matched ERM baseline trains for 300 epochs with learning rate 10^{-3}. The reported steered setting uses the same backbone but a warm-start curriculum with refreshed audits between stages. In the primary joint-pairwise setting we optimize the two pairwise terms and leave the three-way term unoptimized, draw 48 audited anchors per update with equal sampling from the background and co-occurring-object banks, and select checkpoints by the mean validation accuracy over the three official rare-slice subsets (background-conflict, co-occurring-object-conflict, and doubly conflicting). The anchor-bank construction uses `bg_top_pct`$=$ `co_top_pct` $= 40$, keeps smaller alignment buckets down to one tenth of the largest bucket, uses `probe_per_bucket`$= 24$, and enforces `min_obj_mask_frac`$= 0.09$ and `min_obj_effect_bg_anchor`$= 4.0$. The pairwise suppression thresholds are `tau_bg`$= 1.0$ and `tau_co`$= 0.5$.

The steered stages use a 10-epoch learning-rate warmup, no frozen-backbone period, and refreshed audit banks at each stage boundary. Table 5 gives the training schedule.

Table 5: **UrbanCars staged steering schedule.** Stagewise warm-start schedule used for the reported UrbanCars runs. The last column gives the target pairwise interaction level as a fraction of the audited reference magnitude; the same target is used for both optimized interactions within each stage.

Stage	Epochs	LR	BG weight	CoObj weight	Target interaction level	
ERM warm start	20	10^{-3}	–	–	–	–
Steering stage 1	100	4×10^{-5}	0.0020	0.0035	0.90	0.90
Steering stage 2	90	3×10^{-5}	0.0020	0.0035	0.70	0.70
Steering stage 3	90	2×10^{-5}	0.0020	0.0035	0.50	0.50

D.4 PACO

All PACO runs use an ImageNet-pretrained ResNet-50, 224px object crops, AdamW, AMP, and channels-last tensors. ERM trains for 20 epochs with batch size 128, learning rate 2×10^{-4}, weight decay 10^{-4}, natural-image cross-entropy, and a two-epoch frozen-backbone warmup. Steering initializes from the seed-matched ERM checkpoint and fine-tunes for 12 epochs with learning rate 10^{-5}, batch size 128, 32 audited anchors per update, mixed fill during training, score mode true-class margin, positive lower-bound target $K = 0.5$, $\lambda_{\Delta} = 0.05$, $\lambda_{\text{order}} = 0.02$, $\lambda_{\text{wrong}} = 0.01$, and $\lambda_{\text{stability}} = 0.005$. The fine-tuning learning rate is constant; warmup applies only to the target-pair coefficient.

The PACO anchor bank is built from the ERM training audit. We require the ERM model to classify the natural x_{00} crop correctly, then retain anchors whose audited part-pair interaction is weak or below target so that the steering stage focuses on examples with headroom for positive synergy. Edited corners enter the interaction, order, and wrong-class terms; natural crops provide cross-entropy and the ERM-reference stability term.

Checkpoint selection uses the PACO validation manifest. The main PACO tables evaluate the fixed validation-selected checkpoints on a separate PACO-LVIS test manifest of 2,516 object instances that is not used for checkpoint selection. The test manifest uses the same five categories and the same category-specific target part pairs as the training and validation manifests.

We also train a diagnostic PartDrop baseline. PartDrop uses the same ResNet-50 architecture and the same 20-epoch optimizer schedule as ERM, but each training sample is replaced by a randomly sampled PACO corner: the natural x_{00} crop with probability 0.5, and one of x_{10}, x_{01}, or x_{11} with total probability 0.5. Cross-entropy is applied to the object label of the sampled crop. PartDrop uses no finite-difference interaction loss and is selected by natural validation Top-1. It tests whether generic exposure to masked part images is sufficient to reproduce the targeted part-pair mechanism.

D.5 PSG

All PSG learned models use a ResNet-50 backbone with ImageNet initialization for ERM, 224px crops, crop padding 0.15, batch size 128, AdamW, weight decay 10^{-4}, AMP, channels-last tensors, and CUDA. The appendix ERM row is a matched-budget 28-epoch baseline: the same 4-epoch warm start used by the steered model is continued for 24 additional BCE-only epochs so that ERM and steering receive the same total number of optimization epochs. The ERM warm start itself trains for 4 epochs with learning rate 3×10^{-4}, BCE loss on natural relation crops, 12 dataloader workers, and mean-background filling for audit-style removals. The warm steering stage trains from this ERM checkpoint for 16 epochs with learning rate 5×10^{-5}, 32 anchors per update, mixed fill, score mode `ovr_margin`, $K = 0.5$, $\lambda_{\Delta} = 0.05$, $\lambda_{\text{order}} = 0.03$, $\lambda_{\text{neg}} = 0.02$, $\lambda_{\text{stability}} = 0.015$, and warmup fraction 0.25.

The final reported PSG recipe continues from the warm steered checkpoint for 8 epochs with learning rate 10^{-5}, 48 anchor crops per update, mixed-feather filling, blur radius 11, feather radius 4, and true-logit score mode. The target is a banded positive-synergy objective with global $K = 0.20$, upper band 0.45, and predicate-grouped targets $(0.25, 0.16, 0.08)$ for high, medium, and low groups, clipped to $[0.05, 0.35]$. The final loss uses $\lambda_{\Delta} = 0.030$, $\lambda_{\text{order}} = 0.030$, $\lambda_{\text{neg}} = 0.020$, `wrong_top_k`$= 5$, `tau_neg`$= 0.0$, $\lambda_{\text{prior_neg}} = 0.030$, `prior_neg_tau`$= 0.0$, $\lambda_{\text{stability}} = 0.12$,

and $\lambda_{\text{anchor_x11}} = 0.08$ with `anchor_x11_margin`$= 0.0$. Tiny subject/object masks below area fraction `tiny_mask_area_frac`$= 0.01$ are downweighted with `tiny_mask_weight`$= 0.35$. The selected checkpoint view is `best_by_map_floor_synergy`, with validation mAP floor 0.390 and probe batches 16.

E Counterfactual Construction and Rerun Protocol

E.1 Waterbirds

In the released-mask Waterbirds setting, each anchor image provides four corners. x_{00} is the original image, with the bird present on the anchor background. x_{10} removes the bird by inpainting the dataset bird mask on that same anchor background. x_{11} is a donor image sampled uniformly at random from the valid donor pool after the donor bird has been removed with its own mask and inpainted away, so the bird is absent on the alternate background. x_{01} pastes the exact masked bird crop from the anchor image onto this cleansed donor background, so the bird is present under the alternate background. In other words, x_{00}/x_{10} share the anchor background, x_{01}/x_{11} share the donor background, x_{00}/x_{01} keep the bird present, and x_{10}/x_{11} keep it absent. In the final masked pipeline, the donor background is histogram-matched to the anchor image before pasting, and the paste uses the thresholded dataset bird mask. Two distinct donor backgrounds are sampled per anchor when available, and the loss averages the resulting pair of finite differences. This Waterbirds steering setup does not use group labels in training, even though the task loss itself still uses only bird labels on natural images.

The Waterbirds anchor bank is built once from the ERM checkpoint on the training split. For each candidate anchor, we compute the absolute second-order interaction and rank anchors by this score. The reported setting keeps the top 55% of scored anchors and balances the retained bank by class label. In the final reported runs, no additional post-top ranked filtering is applied: `ranked_filter_after_top`$=$ false, `ranked_ebg_abs_max`$= -1.0$, and `ranked_eobj_min`$= -1.0$. The remaining loss-side quality gates are controlled numerically by `tau_bg_min`$= 10.0$, `tau_bg_fac`$= 0.0$, `tau_bg_guard_min`$= 0.5$, `tau_bg_guard_fac`$= 0.02$, `guard_weight`$= 0.13$, `lam_obj`$= 0.6$, and `lam_obj_floor`$= 0.3$. For the released-mask comparison, bank construction uses the training split, checkpoint selection uses validation WGA and group labels, and accuracy and interaction ECDFs use the held-out test split. To rerun the main Waterbirds tables, one should train ERM for 30 epochs for each seed, audit the training split to form this bank, fine-tune for 20 epochs with the settings in Appendix D.2, and evaluate the validation-WGA-selected checkpoint for each seed.

Alternative Waterbirds localization. The frozen-segmenter variant generates bird masks once using a COCO-pretrained Mask R-CNN. It uses no CUB masks or boxes for mask generation, eligibility, interaction ranking, or training, and otherwise follows the released-mask recipe. Reported mask coverage is 99.81%. A second variant derives a class-activation map (Zhou et al., 2016) for the correct bird class from each seed's ERM classifier and converts it to a mask with classical GrabCut (Rother et al., 2004); it uses no segmentation-pretrained model or CUB masks/boxes in mask generation, ranking, or steering. After the standard 256-resize/224-center-crop, the true-class linear-head weights combine the ResNet-50's 7×7 final features (`layer4`); ReLU and per-image min–max normalization precede sure-foreground thresholding at $\mu + 2\sigma$ (peak-cell fallback) and Otsu thresholding for the peak-containing probable-foreground component, dilated by one feature cell. Nearest-neighbor upsampling initializes five GrabCut iterations at 224×224, with the border fixed as background and remaining unassigned pixels as probable background; we retain the peak-containing foreground component (largest-component fallback if the peak is excluded), use the dilated CAM component if GrabCut fails or is empty, and freeze masks during steering. Table 6 reports natural test accuracy and interaction summaries.

Mask prediction uses torchvision's `maskrcnn_resnet50_fpn_v2` with `COCO_V1` weights and built-in preprocessing. Among COCO bird detections passing its default score threshold 0.05, it takes the highest-scoring detection and thresholds the soft mask at 0.5. Missing detections and empty masks exclude an image only from auxiliary corners, not natural-image supervision.

These interaction audits use released CUB masks on test images. Each row statistic is the absolute value of the signed interaction averaged over three fixed donors. These are descriptive audit summaries, not paired comparisons between localization variants.

Table 6: **Waterbirds localization variants.** ResNet-50 results; accuracy is in percent and $|\Delta^2|$ is in raw margin units.

Method / localization	WGA	Average	Mean $\lvert\Delta^2\rvert$
ERM	81.26 ± 0.90	93.03 ± 1.03	1.411 ± 0.059
IS: released CUB masks	89.04 ± 1.01	95.54 ± 0.51	0.557 ± 0.008
IS: frozen Mask R-CNN	89.51 ± 0.09	95.29 ± 0.77	0.536 ± 0.029
IS: ERM CAM + GrabCut	88.11 ± 1.15	94.72 ± 0.54	0.701 ± 0.064

E.2 URBANCARS

UrbanCars uses the fixed four-corner constructions defined by the benchmark: variants differ by whether the target object, the co-occurring object, and the aligned or conflicting background are present. The primary setting steers object×background and object×co-occurring-object; the separate experiment in Appendix I.6 also steers the three-way term in the final stage. At each stage boundary, we audit the current checkpoint on the training split and build separate anchor banks for the two optimized interactions. In the reported setting, bank construction is stratified by alignment bucket: within each bucket, training samples are ranked by absolute interaction magnitude, the top 40% are retained, and small buckets are preserved so that each contains at least one tenth as many samples as the largest bucket. We also require `min_obj_mask_frac`$= 0.09$ and a minimum object-only effect `min_obj_effect_bg_anchor`$= 4.0$ on the anchor background before a sample is admitted to the bank.

During training, the UrbanCars schedule refreshes these audit banks at every stage boundary. Each update draws 48 audited anchors in total, split evenly between the background and co-occurring-object banks. The pairwise penalties use a hinge loss that keeps the current interaction below a stage-dependent fraction of the audited reference magnitude. To rerun this staged UrbanCars setting, one should train the ERM warm start for 20 epochs for each seed, audit the resulting checkpoint, run the $100/90/90$ stagewise steering schedule in Table 5 with refreshed audits between stages, and select the final checkpoint by the mean validation accuracy over the three official rare-slice subsets.

E.3 PACO

For each PACO object crop, the two semantic axes are toggles on selected part groups A and B. We use the anchor-toggle convention from the main paper: x_{00} is the natural crop with both selected parts visible, x_{10} removes part group A while keeping part group B, x_{01} removes part group B while keeping part group A, and x_{11} removes both selected part groups. All other object pixels remain visible. The audited score is the true-class margin, and the pairwise interaction is $\Delta^2_{\mathrm{partA}\times\mathrm{partB}} = m_{11} - m_{10} - m_{01} + m_{00}$.

To rerun the main PACO results, one should first build the targeted PACO train, validation, and test manifests and class map, train the ERM baseline for each seed with the settings in Appendix D.4, audit the training split to form the part-pair anchor bank, and then fine-tune from the seed-matched ERM checkpoint for 12 epochs with the steering hyperparameters above. For each seed, the reported steered model is the checkpoint attaining the highest natural validation Top-1 accuracy, evaluated on the separate PACO-LVIS test manifest. The PartDrop diagnostic baseline is trained from ImageNet initialization with the same 20-epoch optimizer schedule as ERM, using randomly sampled natural and masked PACO corners under ordinary cross-entropy, and is selected by natural validation Top-1.

E.4 PSG

PSG manifests are built from COCO image roots, OpenPSG-style JSON relation annotations, and panoptic PNG masks. The parser uses predicate classes, thing/stuff classes, per-image relations, segment metadata, and panoptic file names to produce natural pair rows and positive anchor rows. The final experiment uses the selected 12 predicates with minimum predicate count filtering, multi-hot pair labels, and raw zero-based predicate IDs internally. For each relation pair, the crop is the padded union of the subject and object masks. The four PSG corners use the same anchor-toggle convention

as the main paper: x_{00} is the natural subject/object crop, x_{10} removes the subject while keeping the object, x_{01} removes the object while keeping the subject, and x_{11} removes both entities.

Mask loading prioritizes panoptic PNGs with local RGB-to-ID conversion. In the reported runs, missing or ambiguous masks are skipped rather than synthesized. The final steering anchors come from the hard-plus-global anchor bank built from audited training rows, with additional guards for wrong predicates and category-prior-favored predicates. To rerun the PSG appendix diagnostic, one should build the manifests, train the 4-epoch ERM warm start for each seed, audit to form the warm anchor bank, run the 16-epoch warm steering stage, continue for the 8-epoch soft-low-K stage in Appendix D.5, and evaluate the selected `best_by_map_floor_synergy` checkpoints against the seed-matched, 28-epoch ERM baseline.

F Model Selection, Seeds, and Statistical Reporting

Waterbirds selects baseline and steered checkpoints by validation WGA, which uses validation group labels even when training does not. UrbanCars selects the staged steered checkpoints by mean validation accuracy over the three official rare-slice subsets. PACO uses highest natural validation Top-1 accuracy for ERM, PartDrop, and IS. PartDrop and IS share part annotations, corner types, and the selection metric, but differ in training budget (Appendix D.4). PACO evaluates the selected checkpoints on the separate held-out test manifest, except for the explicitly labeled validation fill-mode diagnostics; the test manifest is not used for selection.

G Compute Resources

All reported runs use one CUDA GPU with AMP. Table 7 summarizes approximate training times and reported-run budgets inferred from the logs; these estimates exclude additional exploratory runs.

Corner preparation can be cached offline; anchor-bank audits and up to 2^k corner evaluations per selected interaction add training cost. Natural-image inference uses only the classifier. Edit operators are specified in Appendix E.

Table 7: **Approximate training cost for the final seed results.** Times are estimates inferred from experiment logs.

Benchmark family	Approx. wall-clock	Reported-run budget
Waterbirds	steered fine-tune ≈54–56 min/seed; ERM warm start shorter	3 seeds/backbone for main tables; baseline reruns add compute
UrbanCars	each steered 90–100 epoch stage ≈3.7–3.9 h; full staged steering setting ≈12 GPU-h/seed	3 matched ERM seeds + 3 steered seeds
PACO	ERM ≈20 epochs/seed; 12-epoch steering continuation under 1 GPU-h/seed	3 ERM + 3 IS + 3 target-loss controls
PSG appendix	full warm-start + steering path on the order of several GPU-h/seed; final 8-epoch continuation is shorter	3 matched ERM seeds + 3 steered seeds

H Assets, Licenses, and Terms of Use

Table 8 lists the central external assets used in this paper, the original owners we credit, the source used to obtain them, and the corresponding license or terms-of-use statement. We use these assets for non-commercial academic research and do *not* redistribute dataset archives or pretrained checkpoints as part of this submission.

Table 8: **External assets used in the paper.** We cite the original source for each asset and report the license or terms used for this non-commercial academic submission.

Asset	Official citation / owner	Official source or repository	License / terms status	Redistribution in this paper
Waterbirds benchmark	Waterbirds benchmark of Sagawa et al. (2020), derived from CUB-200-2011 Wah et al. (2011) images/segmentations and Places Zhou et al. (2018) backgrounds	Official `group_DRO` Waterbirds tarball; WILDS mirror used only as an alternate access path	`group_DRO` code/release: MIT license. CUB-200-2011 images are restricted to non-commercial research/education use; segmentations are separately distributed by the CUB maintainers. Places image copyrights remain with the image owners.	No dataset archive redistributed; only aggregate metrics and reduced-size illustrative panels in the manuscript
UrbanCars benchmark	UrbanCars / Whac-A-Mole benchmark of Li et al. (2023)	Official Whac-A-Mole release	Creative Commons Attribution-NonCommercial 4.0 International (CC BY-NC 4.0)	No dataset archive redistributed; only aggregate metrics and audit plots
COCO images and panoptic masks	Microsoft COCO dataset of Lin et al. (2014)	Official COCO release	COCO annotations and website: Creative Commons Attribution 4.0 (CC BY 4.0). COCO does not own the image copyrights; image use must follow the source Flickr terms.	No COCO archive redistributed; only aggregate metrics and reduced-size qualitative examples
PACO / PACO-LVIS annotations	PACO dataset of Ramanathan et al. (2023)	Official PACO GitHub repository and PACO-LVIS annotation download	PACO repository code is MIT licensed. PACO-LVIS uses COCO train/val images, so image use follows the COCO/Flickr terms above; we use the official annotation release and do not redistribute it here.	No annotation archive redistributed; only aggregate metrics and reduced-size illustrative examples
PSG / OpenPSG annotations	Panoptic Scene Graph Generation dataset of Yang et al. (2022)	Official `Jingkang50/OpenPSG` release	OpenPSG repository: MIT license. Underlying image use follows COCO/Flickr terms as above.	No annotation archive redistributed; only aggregate metrics and reduced-size qualitative examples
DINOv2 ViT-B/14 weights	Official DINOv2 pretrained release of Oquab et al. (2024) (Meta AI)	Meta AI official DINOv2 repository	DINOv2 code and model weights: Apache License 2.0	No pretrained weights redistributed
ResNet-50 (He et al., 2016), pretrained on ImageNet (Deng et al., 2009)	Torchvision / ImageNet-pretrained ResNet-50 release	Official `pytorch/vision` model zoo	Torchvision repository: BSD 3-Clause License; pretrained weights used through the official model zoo	No pretrained weights redistributed

I Additional Experimental Results

I.1 Colored MNIST

We include Colored MNIST (Arjovsky et al., 2019) here as a compact illustration of the first-order special case of our framework. We construct a variant where the digit color is strongly correlated with a noisy binary label during training ($\rho = 0.95$) and flipped at test time ($\rho = 0.05$). In this setting, the desired logic is invariance to color, so we suppress the first-order effect $\Delta^1(\text{color})$ using a simple channel-swap counterfactual.

Table 9 shows the expected pattern. The ERM baseline exploits the color shortcut, achieving high IID accuracy but collapsing on the OOD split. Steering the first-order color effect toward zero closes this generalization gap and recovers performance near the digit-only accuracy ceiling of roughly 75%, which comes from the 25% label-noise rate in this Colored MNIST construction.

Table 9: **Colored MNIST results.** Suppressing the first-order color effect closes the IID–OOD generalization gap.

Method	λ	IID Acc. (%)	OOD Acc. (%)
ERM	0.0	95.00	5.19
Steered	0.5	75.53	71.36
Steered	0.7	74.30	**73.37**
Steered	0.9	74.08	73.14

I.2 WATERBIRDS: FULL BACKBONE TABLES

The main paper reports a compact two-backbone summary. For completeness, we include here the full per-group tables for the ResNet-50 and DINOv2 ViT-B/14 runs used in the Waterbirds section.

Table 10: **Waterbirds test results with ResNet-50 (3 seeds).** Avg. Acc. and WGA are mean±std in percentage points. Group columns show seed-mean accuracy in percent.

Method	WGA	Avg. Acc.	ΔWGA	g_0	g_1	g_2	g_3
Baselines without Training Group Labels							
ERM	81.26±0.90	93.03±1.03	0.00±0.00	99.72	89.28	81.26	94.44
Focal Loss Lin et al. (2017)	79.94±1.85	91.65±1.36	-1.31±2.10	99.25	85.84	80.89	96.16
CVaR Levy et al. (2020)	82.45±2.65	92.47±0.69	1.19±3.54	99.41	87.20	82.66	96.42
Spectral Decoupling Pezeshki et al. (2021)	82.65±1.54	92.29±1.51	1.39±1.42	99.45	86.40	83.80	96.31
JTT Liu et al. (2021)	79.54±2.07	90.62±2.36	-1.71±2.43	99.17	83.34	79.54	97.20
Interaction Steering							
Steered	**89.04±1.01**	**95.54±0.51**	7.79±0.47	99.45	93.50	89.04	95.48
With Group Labels							
GroupDRO Sagawa et al. (2020)	88.42±1.13	94.82±0.53	7.17±1.09	98.82	92.28	88.42	96.11

Table 11: **Waterbirds test results with DINOv2 ViT-B/14 (3 seeds).** Avg. Acc. and WGA are mean±std in percentage points. Group columns show seed-mean accuracy in percent.

Method	WGA	Avg. Acc.	ΔWGA	g_0	g_1	g_2	g_3
Baselines without Training Group Labels							
ERM	89.55±1.91	94.96±0.93	0.00±0.00	99.81	90.36	90.50	98.55
Focal Loss Lin et al. (2017)	88.29±3.03	95.48±0.50	-1.26±3.96	99.72	92.11	88.94	98.96
CVaR Levy et al. (2020)	89.40±0.99	95.16±0.85	-0.15±2.77	99.85	91.01	90.13	98.29
Spectral Decoupling Pezeshki et al. (2021)	88.32±2.72	95.65±0.87	-1.23±1.90	99.82	92.77	88.42	98.34
JTT Liu et al. (2021)	86.72±2.07	93.38±0.76	-2.83±3.48	99.62	87.18	87.44	99.17
Interaction Steering							
Steered	**94.81±0.48**	**97.72±0.31**	5.26±2.17	99.42	97.16	94.81	96.63
With Group Labels							
GroupDRO Sagawa et al. (2020)	94.01±0.69	96.86±0.79	4.46±2.49	99.35	94.78	94.39	97.92

I.3 WATERBIRDS: LAST-LAYER RETRAINING

We evaluate whether IS complements deep feature reweighting (DFR) (Kirichenko et al., 2023) using ResNet-50 ERM and IS checkpoints for seeds $1, 21, 42$. We freeze each backbone and replace its classifier with an ℓ_1-regularized logistic-regression head fitted using the official DFR implementation. Features are extracted without augmentation, using resize-256 and center-crop-224 preprocessing. Refitting uses equal numbers of examples from the four ground-truth bird-label×background groups, with equal class weights.

DFR(Val) selects the regularization parameter C from $\{1, 0.7, 0.3, 0.1, 0.07, 0.03, 0.01\}$. We randomly split the validation set in half, fit the head on a group-balanced sample from one half, and select C by WGA on the other. During tuning, feature standardization is fitted on the balanced refitting sample. For the final classifier, standardization statistics are estimated from training-split features, and we average the coefficients and intercepts of 20 heads fitted on balanced samples of the full validation set.

ERM and IS use the same validation split and balanced samples within each seed. Validation WGA also selects the base checkpoints; the test split is used only for final evaluation. Results are mean±sample standard deviation across three seeds, and Avg. denotes plain test accuracy.

Table 12: **Waterbirds with validation-set DFR.** ResNet-50 results are mean±sample standard deviation over three seeds.

Method	WGA (%)	Avg. (%)
ERM	81.26 ± 0.90	93.03 ± 1.03
ERM + DFR (Val)	91.90 ± 1.02	94.49 ± 0.30
IS	89.04 ± 1.01	$\mathbf{95.54 \pm 0.51}$
IS + DFR (Val)	$\mathbf{92.52 \pm 0.68}$	95.46 ± 0.66

I.4 Waterbirds: ResNet-50 ExMap Comparison

We also ran a ResNet-50 comparison to ExMap Chakraborty et al. (2024), an explanation-based pseudo-group method that does not use training group labels. This run follows the official ExMap ingredients as closely as possible in our Waterbirds pipeline: zennit LRP explanations on the validation split, spectral clustering with automatic eigengap selection, and DFR-style L1 logistic-regression retraining on frozen ERM features using the inferred pseudo-groups. The official ExMap LRP implementation is ResNet-oriented, so this comparison uses ResNet-50.

Table 13: **Waterbirds ResNet-50 comparison to official-style ExMap.** ExMap uses zennit LRP, spectral clustering with eigengap selection, and DFR-style L1 logistic-regression retraining on validation pseudo-groups. Avg. Acc., WGA, and ΔWGA are mean±std over three seeds in percentage points; group columns show seed-mean accuracy in percent.

Method	**WGA**	**Avg. Acc.**	**ΔWGA**	g_0	g_1	g_2	g_3
ERM	81.26±0.90	93.03±1.03	0.00±0.00	99.72	89.28	81.26	94.44
ExMap Chakraborty et al. (2024)	86.67±1.92	93.55±2.51	5.41±1.51	97.17	90.37	90.21	95.39
Interaction Steering	**89.04±1.01**	**95.54±0.51**	**7.79±0.47**	**99.45**	**93.50**	89.04	**95.48**

ExMap improves ResNet-50 worst-group accuracy substantially over ERM, from 81.26 ± 0.90 to 86.67 ± 1.92, confirming that explanation-derived pseudo-groups are a meaningful no-training-group-label baseline on Waterbirds. Interaction Steering remains higher in this matched comparison, reaching 89.04 ± 1.01 WGA and higher average accuracy.

I.5 Waterbirds: Mechanism Comparison with Fully Counterfactual Augmentation

Full counterfactual augmentation (CF-Aug) uses the same masks and generator as IS to add labeled bird–background compositions to cross-entropy training, increasing exposure to the scarce conflict groups. IS keeps cross-entropy on natural images and uses edited corners only in auxiliary losses. We compare benchmark accuracy, interaction magnitude, and predictions on birdless donor backgrounds. These probes have no meaningful correct bird-class label: their alignment with the background-associated class, confidence, and entropy measure how readily the classifier substitutes scene context for bird evidence.

Full CF-Aug has the highest WGA in Table 14, while IS has the smallest interaction magnitude and the weakest background dependence on the seed-42 birdless probes. Thus balancing labeled compositions and targeting their interaction can produce different score behavior even when both improve benchmark accuracy.

I.6 UrbanCars: Full Metrics and Interaction Audits

Figure 3 compares all three gaps and shows the single-target tradeoffs. Table 15 gives the complete metrics and standard deviations; Figure 5 combines the pairwise and unoptimized three-way audits.

The following audit concerns the primary joint-pairwise setting, whose three-way term is unoptimized. As an additional audit, we measure the unoptimized three-way interaction $\Delta^3_{\text{obj}\times\text{bg}\times\text{co}}$ on held-out

Table 14: **Waterbirds mechanism comparison.** Top: three-seed ResNet-50 mean±std. Bottom: seed-42 probes formed by removing donor birds. Lower background-class alignment and confidence, and higher entropy, indicate weaker background dependence.

Question / metric	ERM	Steered	Full CF-Aug
Main benchmark metrics (3 seeds)			
Worst-group accuracy (%, ↑)	81.26±0.90	89.04±1.01	**92.52±0.56**
Average accuracy (%, ↑)	93.03±1.03	95.54±0.51	**96.45±0.54**
Mean absolute interaction on test audits ($\lvert\Delta^2\rvert$, ↓)	1.411±0.059	**0.557±0.008**	0.772±0.091
Birdless background-only diagnostics (seed 42)			
On birdless donor backgrounds, how often does the top prediction match the class associated with the visible background? (↓)	0.917	**0.587**	0.668
On birdless donor backgrounds, how much probability goes to the class associated with the visible background? (↓)	0.779	**0.527**	0.561
On birdless donor backgrounds, how confident is the model overall when no bird is present? (↓)	0.803	**0.562**	0.607
On birdless donor backgrounds, how uncertain is the model when no bird is present? (↑)	0.452	**0.672**	0.646

Table 15: **UrbanCars official benchmark metrics.** Paper-reported baselines are included for context. Our matched ERM and steering rows are mean±std over three seeds. Gap values closer to zero are better.

Method	ID Acc	BG Gap	CoObj Gap	BG+CoObj Gap
Baseline				
ERM (paper)	97.60	-15.30	-11.20	-69.20
Data Augmentation / Regularization				
Mixup	98.30	-12.60	-9.30	-61.80
CutMix	96.60	-45.00	-4.80	-86.50
Cutout	97.80	-15.80	-10.40	-71.40
AugMix	98.20	-10.30	-12.10	-70.20
SD	97.30	-15.00	-3.60	-36.10
CF+F Aug	96.80	-16.00	0.40	-19.40
Shortcut Discovery / Reweighting				
LfF	97.20	-11.60	-18.40	-63.20
JTT (E=1)	95.90	-8.10	-13.30	-40.10
EIIL (E=1)	95.50	-4.20	-24.70	-44.90
JTT (E=2)	94.60	-23.30	-5.30	-52.10
EIIL (E=2)	95.50	-21.50	-6.80	-49.60
DebiAN	98.00	-14.90	-10.50	-69.00
Benchmark-Specific Multi-Shortcut Method				
LLE (paper)	96.70	-2.10	-2.70	-5.90
Matched Setup				
ERM (ours, 3 seeds)	97.84±0.23	-13.84±2.00	-10.77±1.84	-71.71±2.20
BG-only steer (ours, 3 seeds)	97.49±0.75	-4.29±1.14	-13.62±0.54	-37.75±1.12
Co-only steer (ours, 3 seeds)	97.85±0.02	-18.78±0.91	-4.25±0.71	-50.51±4.07
Joint steering (ours, 3 seeds)	95.10±0.37	-3.77±0.56	-4.83±0.56	-16.57±1.63

anchors. For three binary factors A, B, C over the eight corners f_{abc}, this quantity is

$$\Delta^3_{A\times B\times C} = f_{111} - f_{110} - f_{101} - f_{011} + f_{100} + f_{010} + f_{001} - f_{000}. \tag{22}$$

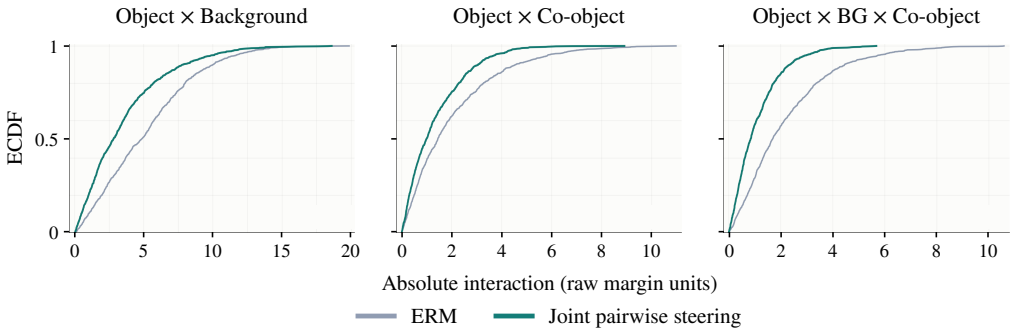


Figure 5: **UrbanCars interaction audits under joint pairwise steering.** Held-out ECDFs of absolute interactions. Left and middle: the two optimized pairwise terms. Right: a representative audit of the unoptimized three-way term, which also contracts. Direct third-order suppression is evaluated separately below.

Figure 5 (right) shows a representative ECDF of its absolute magnitude. Pairwise steering therefore also reduces the unoptimized third-order term on these held-out anchors.

UrbanCars direct third-order suppression. Each run starts from its stage-2 checkpoint and replaces the final 90 epochs of the 300-epoch curriculum with joint pairwise and third-order steering. We select the third-order weight 0.03 from $\{0.02, 0.03, 0.04\}$ using the first run's validation results and use the same training settings for all three seeds. Checkpoints maximize mean validation accuracy over the three official rare slices. Test behavior and the final 1,000-example interaction audit do not select checkpoints. Table 16 reports behavior and third-order interaction.

Table 16: **UrbanCars third-order experiment.** Entries are mean±sample standard deviation. Test accuracy is in percent; gaps are in percentage points (closer to zero is better). Mean $|\Delta^3|$ is measured on the matched validation audit.

Setting	ID accuracy	BG gap	CoObj gap	BG+CoObj gap	Mean $\lvert\Delta^3\rvert$
Joint pairwise steering	95.10 ± 0.37	-3.77 ± 0.56	-4.83 ± 0.56	-16.57 ± 1.63	1.092 ± 0.017
Pairwise + third-order	95.06 ± 0.70	-3.86 ± 1.26	-5.59 ± 1.10	-15.33 ± 2.17	0.970 ± 0.013

I.7 PACO: Held-Out Test, Controls, Specificity, and Fill-Mode Diagnostics

This appendix reports held-out target-pair behavior, the PartDrop comparison, the target-pair loss control, specificity to the named pair, and fill-mode diagnostics. Models are selected on PACO validation data and evaluated on separate PACO-LVIS test instances. Percentage quantities are in points; interactions and margin drops are in raw true-class margin units.

Table 17: **PACO held-out target-pair behavior.** Fixed validation-selected checkpoints are evaluated on the separate PACO-LVIS test manifest. Percentage quantities are reported in points; margin quantities are raw true-class margin units.

Method	Natural Top-1	Mean Δ^2	$\Delta^2 > K$	Full-corner dom.	Drop A	Drop B	Wrong Δ^2
ERM	97.66±0.38	−0.382±0.041	6.77±1.95	35.35±2.61	0.347±0.087	0.371±0.125	0.083±0.030
Interaction Steering	97.79±0.12	0.339±0.029	34.74±1.19	79.19±2.43	0.816±0.038	0.806±0.052	−0.139±0.008
Steered–ERM	+0.13±0.26	+0.720±0.065	+27.97±1.90	+43.84±4.55	+0.469±0.115	+0.434±0.159	−0.223±0.031

The held-out target-pair table above shows that the validation-selected steered checkpoints preserve natural classification on held-out PACO-LVIS test instances while substantially increasing the intended true-class part-pair interaction. Natural Top-1 changes by only +0.13 points, but mean Δ^2 shifts from −0.382 to +0.339, the fraction above K increases by +27.97 points, and full-corner dom-

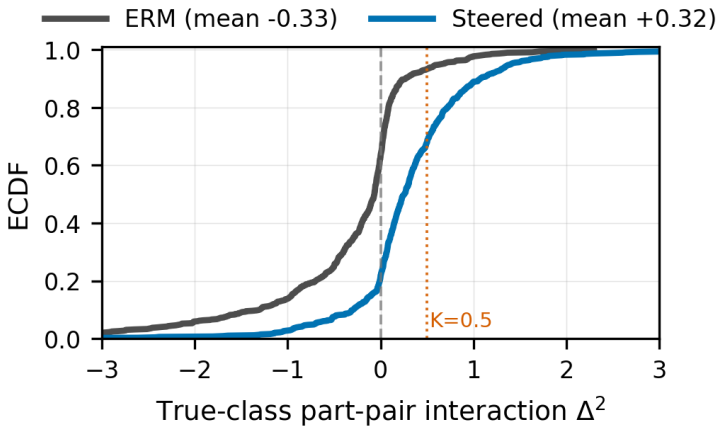


Figure 6: **PACO part-pair interaction ECDF.** For a representative seed, Interaction Steering shifts the audited true-class $\Delta^2_{\text{partA}\times\text{partB}}$ distribution toward the positive target threshold. The corresponding main-text corner stencil appears in Figure 4.

inance increases by $+43.84$ points. Removing either selected part also produces a larger true-class margin drop after steering, while the top wrong-class interaction decreases.

Table 1 gives the full PartDrop comparison. PartDrop improves masked-corner accuracy while reducing sensitivity to the selected pair; IS preserves natural accuracy while increasing that pair's margin contribution and interaction. The control shares annotations and corner types, with its training and selection protocol specified in Appendices D.4 and F.

Target-pair loss control. We compare full IS with a control that sets $\lambda_\Delta = 0$. Each of the three seeds uses the same saved ERM initialization, seed-specific training bank, 12-epoch fine-tuning budget, and remaining hyperparameters (Appendix D.4). Natural-image cross-entropy and the order, wrong-class, and stability guards remain active, including the wrong-class finite differences, auxiliary corner forwards, BatchNorm updates, and mixed-fill sampling. Both arms use natural validation Top-1 for checkpoint selection: selected epochs are $8/12/11$ for full IS and $2/3/2$ for the control. Both are evaluated with the same evaluator and settings on $2{,}516$ held-out target-pair anchors per seed, using identical images, masks, and deterministic mean-background fills. Table 18 reports arm means and paired full-IS-minus-control changes. The 95% intervals use the $50{,}000$ crossed seed-and-anchor bootstrap replicates described in Appendix I.7, with shared anchor draws across seeds and arms.

Table 18: **PACO target-pair loss component control.** Both arms retain all guards and use the same fine-tuning budget and validation-selection rule. Arm columns are means over three seeds; the last column gives paired changes and 95% bootstrap intervals. Percentage metrics and their changes are in points.

Metric	Without target-pair loss	Full IS	Paired change [95% CI]
Natural Top-1 (%)	97.99	97.79	-0.20 $[-0.56, +0.15]$
Mean target Δ^2	-0.102	$+0.339$	$+0.441$ $[0.408, 0.479]$
$\Delta^2 > 0.5$ (%)	8.89	34.74	$+25.85$ $[24.15, 27.58]$
Full-corner dominance (%)	78.68	79.19	$+0.50$ $[-1.11, +2.00]$

The target-pair term increases positive interaction even though the natural corner already dominates for most control anchors. Its incremental effect is therefore distinct from preserving natural-corner ordering. The natural Top-1 difference is -0.20 points, with a 95% interval of $[-0.56, +0.15]$.

The incremental mean interaction changes are $+0.092$ (CI $[0.079, 0.105]$) for mixed-A pairs and $+0.138$ (CI $[0.112, 0.162]$) for mixed-B pairs on $1{,}915$ anchors, and $+0.039$ (CI $[0.030, 0.049]$) for

fully off-target pairs on 1,070 anchors. On common anchors, the target change exceeds the larger mixed change by 0.267 (CI $[0.243, 0.292]$) and the off-target change by 0.313 (CI $[0.284, 0.343]$). The larger mixed change is recomputed within each bootstrap replicate. These contrasts support preferential control of the named pair under the retained guards and matched fine-tuning budget.

Table 19: **PACO off-target pairwise interaction audit.** Each row builds a new four-corner audit on the held-out PACO-LVIS test manifest. Mixed pairs share one steered part with one non-target part; fully off-pairs contain two non-steered parts. The target pair changes most, mixed pairs move moderately, and fully off-target pairs remain near zero.

Audited pair type	Rows	ERM mean Δ^2	PartDrop mean Δ^2	Steered mean Δ^2	Steered–ERM	ERM $\Delta^2{>}K$	Steered $\Delta^2{>}K$	Steered corner-top
Target steered pair	2516	-0.382 ± 0.041	-0.055 ± 0.026	0.339 ± 0.029	$+0.720\pm0.065$	6.77	34.74	79.19
Mixed-A pair	1915	-0.172 ± 0.063	-0.035 ± 0.010	0.027 ± 0.020	$+0.200\pm0.044$	3.46	8.32	76.28
Mixed-B pair	1915	-0.172 ± 0.066	-0.031 ± 0.014	0.121 ± 0.040	$+0.293\pm0.038$	5.33	16.10	74.19
Fully off-pair	1070	-0.037 ± 0.016	-0.032 ± 0.019	0.031 ± 0.015	$+0.068\pm0.003$	2.24	4.30	67.13

Table 19 shows the largest interaction change on the target pair, intermediate changes on mixed pairs, and near-zero fully off-target interactions. The paired bootstrap below tests these contrasts on common eligible anchors.

Paired uncertainty for specificity. We use 50,000 crossed clustered bootstrap replicates with NumPy's `default_rng(20260725)`. Each replicate independently samples three training seeds and the original number of eligible object anchors, uniformly with replacement; one anchor draw is shared across sampled seeds and compared conditions, and all sampled seed–anchor entries receive equal weight. An anchor is identified by image ID, object-instance ID, and source target-pair ID. Each entry is the matched change $\Delta^2_{\mathrm{IS}} - \Delta^2_{\mathrm{ERM}}$. The target change uses 2,516 anchors; target-minus-mixed and target-minus-off-target use 1,915 and 1,070 common anchors, respectively. In each mixed contrast, we subtract the larger of the two mixed-condition means from the target mean. Table 20 reports percentile 95% intervals: the 2.5th and 97.5th percentiles of the replicate estimates, using NumPy's linear interpolation. These bootstrap intervals quantify uncertainty across both seeds and anchors, whereas the three-seed standard deviations summarize variation across training runs.

Table 20: **PACO specificity uncertainty.** Changes and paired contrasts in raw true-class-margin interaction units. The last two rows use common eligible anchors and need not equal differences of full-population row means in Table 2.

Quantity	Estimate	95% CI
Target-pair change	$+0.720$	$[0.647, 0.790]$
Target change − larger mixed change	$+0.356$	$[0.301, 0.419]$
Target change − off-target change	$+0.454$	$[0.355, 0.576]$

Table 21: **PACO non-target part-removal diagnostic on an eligible held-out test subset.** This table uses 1,915 of 2,516 held-out test anchors, restricted to examples with at least one removable non-target annotated part. Natural Top-1 is therefore subset accuracy, not the full-test accuracy in Table 17.

Method	Natural Top-1	Non-target removed Top-1	Lost-correct rate	Non-target drop
ERM	97.98 ± 0.39	97.18 ± 0.86	1.17 ± 0.58	0.085 ± 0.058
PartDrop	97.89 ± 0.25	96.94 ± 0.79	1.34 ± 0.56	0.104 ± 0.059
Interaction Steering	$\mathbf{98.40\pm0.13}$	$\mathbf{98.05\pm0.21}$	$\mathbf{0.63\pm0.10}$	0.342 ± 0.049

Table 21 is a robustness diagnostic rather than a specificity metric. The steered model has a larger margin drop when a non-target annotated part is removed, but these parts are not necessarily irrelevant; car bumpers, chair legs, and bicycle saddles can still provide real object evidence. The accuracy-based metrics are more important here: Interaction Steering has the highest non-target-removed Top-1 and the lowest lost-correct rate, so the increased semantic part sensitivity does not make the classifier brittle to removing a non-target part.

Tables 22–23 test sensitivity to the masking operator on the validation manifest with fixed checkpoints. Entries are paired steered-minus-ERM changes; the natural-corner value -0.24 denotes a 0.24-point

decrease in validation Top-1. Interaction and full-corner-dominance gains persist across mean-background, blur, black, and mixed fills.

Table 22: **PACO validation-manifest masked-corner Top-1 diagnostic under alternate fills.** Entries are paired steered-minus-ERM Top-1 changes on the validation manifest, reported in points for fixed checkpoints.

Fill operator	Natural ΔTop-1	One-part ΔTop-1	Both-parts ΔTop-1
Mean background	-0.24	$+2.03$	$+8.46$
Blur	-0.24	$+0.81$	$+3.87$
Black	-0.24	$+2.51$	$+6.98$
Mixed	-0.24	$+2.10$	$+9.76$

Table 23: **PACO validation-manifest fill-mode robustness diagnostic.** Checkpoints are fixed; only the corner edit operator changes. Percentage columns are paired steered-minus-ERM changes in points; margin quantities are raw true-class margin units. Positive full-corner-dominance and Δ^2 changes persist under blur and black fills.

Fill	Natural ΔTop-1	Mean Δ^2	$\Delta^2 > K$	Full-corner dom.	Full-vs-best	Wrong Δ^2
Mean background	-0.24	$+0.6481$	$+25.01$	$+42.99$	$+0.3618$	-0.2103
Blur	-0.24	$+0.3607$	$+13.49$	$+49.12$	$+0.3697$	-0.1125
Black	-0.24	$+0.6345$	$+30.03$	$+43.42$	$+0.4368$	-0.2380
Mixed	-0.24	$+0.7392$	$+24.39$	$+45.77$	$+0.4166$	-0.2348

These audits evaluate controlled part removals on the five-category subset; generalization to natural occlusion or unseen categories remains untested.

I.8 PSG: Relation-Synergy Diagnostic

The PSG experiment tests subject–object synergy in a setting with strong category-pair priors. We compare ERM and IS at a matched training budget and evaluate both predicate prediction and interaction behavior.

Table 24: **PSG vision-only relation steering.** The category-pair prior uses only subject/object categories and has no image-dependent corner response. ERM and Interaction Steering are ResNet-50 visual classifiers over 12 predicates and three seeds; the ERM row is the matched-budget 28-epoch BCE-only baseline, prediction columns are percentages, Δ^2 and Wrong guard are raw finite-difference quantities, and multi-label F1 binarizes logits at threshold 0.

Method	mAP	Micro F1	Macro F1	Mean Δ^2	$\Delta^2 > 0.5$	Natural-corner top	Wrong guard
Category-pair prior	**60.8**	**71.7**	**54.4**	0.000	0.0	N.A.	N.A.
ResNet-50 ERM	39.1±0.4	51.7±0.8	39.7±0.7	0.009 ± 0.044	30.4±1.2	36.2±1.3	0.790±0.065
Interaction Steering	39.8±0.6	52.4±0.4	39.0±0.7	**0.225±0.057**	29.0±5.4	**41.7±0.4**	**0.122±0.021**

Among learned visual models, steering gives only a modest aggregate prediction change: mAP rises from 39.1 to 39.8, micro F1 rises from 51.7 to 52.4, and macro F1 decreases slightly from 39.7 to 39.0. The mechanism shift is much larger. The mean true-predicate interaction increases from 0.009 to 0.225, natural-corner top rises from 36.2% to 41.7%, and the wrong-predicate guard drops sharply from 0.790 to 0.122. At the same time, the category-pair prior remains much stronger on aggregate prediction.

I.9 PSG Qualitative Corners

Figure 7 shows two audited relation anchors and their four PSG corners.

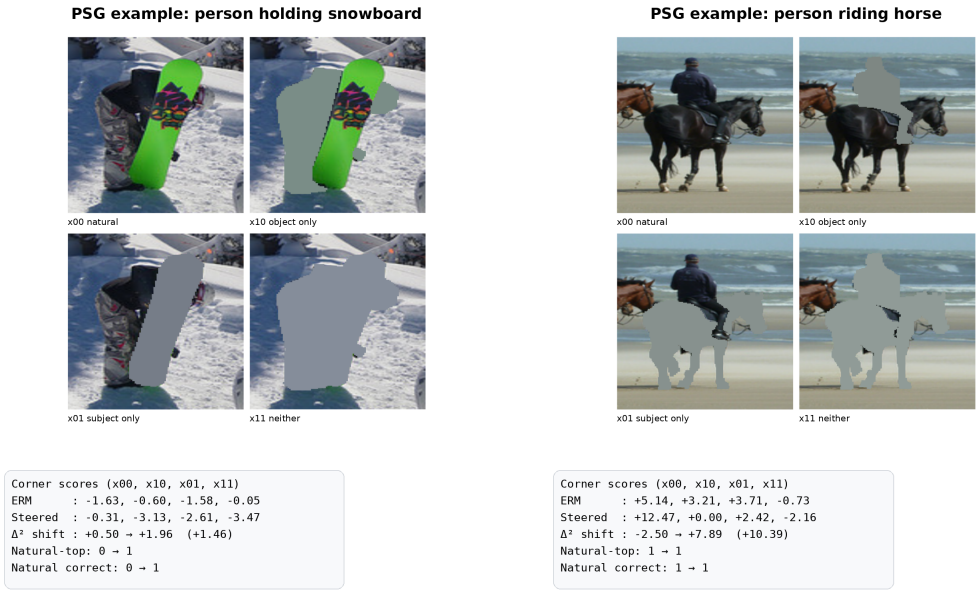


Figure 7: **PSG qualitative corner examples.** Each panel shows four relation-crop corners and true-predicate scores for the 4-epoch ERM warm start and IS. These examples are illustrative; matched-budget results appear in Table 24.

J Sparse Recovery for Larger Factor Sets

Fix a model, an anchor image x, and n semantic factors with a consistent binary corner construction. For $T \subseteq [n]$, let x_T toggle the factors in T, with the remaining factors held at their anchor states, and define $g_x(T) = f(x_T)$. The interaction over a nonempty set S is

$$\Delta_S^{|S|} f(x) = \sum_{T \subseteq S} (-1)^{|S|-|T|} g_x(T) = \mu_x(S), \tag{23}$$

where $\mu_x(S)$ is the Möbius coefficient of g_x. This recovers our second- and third-order stencils. Thus, interactions across all subsets form the Möbius transform of the corner-score function. Computing the full transform directly requires 2^n corner evaluations.

Sparse recovery offers a possible alternative. Let s denote the number of nonzero coefficients and t their maximum interaction order. Under the random-support, coefficient, and scaling assumptions of their Theorem 5.2, Kang et al. (2024) obtain $O(st \log n)$ score evaluations and $O(s \operatorname{poly}(n))$ recovery time. Their method combines structured subsampling, group testing, and iterative subtraction of recovered coefficients; a noise-robust variant is analyzed under a specified measurement-noise model. The savings arise from recovering a sparse collection of interactions jointly, while preserving the finite-difference quantity being measured. Adapting this recovery procedure to our visual edits and steering losses remains future work.