Causal Gradient Steering: Exposing Shortcut Gradients by Destroying Causal Signal

Ahmed Radwan **Ahmad Abdel-Qader** **Mahmoud Soliman**
Omar Abdelaziz **Ahmed Elgazwy** **Mohamed Shehata** **Shan Du**

Department of Computer Science
University of British Columbia
3333 University Way, Kelowna, BC, Canada

ahmed.radwan@ubc.ca aabdelqa@student.ubc.ca mosama97@student.ubc.ca
oabdelaz@student.ubc.ca ahmed.elgazwy@ubc.ca
mohamed.sami.shehata@ubc.ca shan.du@ubc.ca

Abstract

It is often easier to destroy the causal content of a training sample than to isolate it. We use this observation to expose shortcut gradients during training. If an intervention removes the semantic evidence that justifies a sample's label while preserving nuisance structure, then any predictive gradient the model still produces on the destroyed view points toward features that can support the label without the causal signal. The destroyed view is constructed from the same sample as the clean view, and the implemented optimizer estimates this shortcut direction on the current minibatch and applies the steering step periodically. Thus CGS injects a recurring within-sample shortcut-correction signal while remaining domain-label-free and close to ERM in cost. We formalize this principle through a nuisance-only Distributionally Robust Optimization geometry, in which an adversary may perturb nuisance factors while holding the semantic component fixed. The resulting local Danskin majorizer induces a first-order robust-feasible cone in parameter space, and Causal Gradient Steering (CGS) is the closed-form projection of the ERM step onto a tractable single-normal approximation of this cone. We further prove target-risk and probe-admissibility guarantees, showing that exact counterfactual generation in pixel space is unnecessary. CGS is compatible with standard first-order training and can act either as a standalone optimizer or as a plug-in correction for existing DG algorithms. It obtains the strongest average among the listed standard domain generalization benchmark results under the stated DomainBed-style protocol. Moreover, using a single locked CGS hyperparameter configuration, it improves seventeen prior DG methods by up to +15.8 percentage points without method-specific retuning, improves OSTrack-256 across GOT-10k, UAV123, and OTB2015, and improves the matched-protocol SCSD baseline for domain-generalized semantic segmentation from 51.42 to 52.35 mIoU on GTAV+SYNTHIA→{Cityscapes, BDD, Mapillary} and from 35.66 to 36.99 mIoU on GTAV→ACDC. Beyond this algorithm, causal signal destruction defines a research direction parallel to data augmentation. The destruction operator is a design axis, and task-specific destroyers can turn shortcut suppression into a reusable tool for broader vision problems. Code is available at `https://github.com/AhmedMostafaSoliman/CausalGradientSteering`.

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

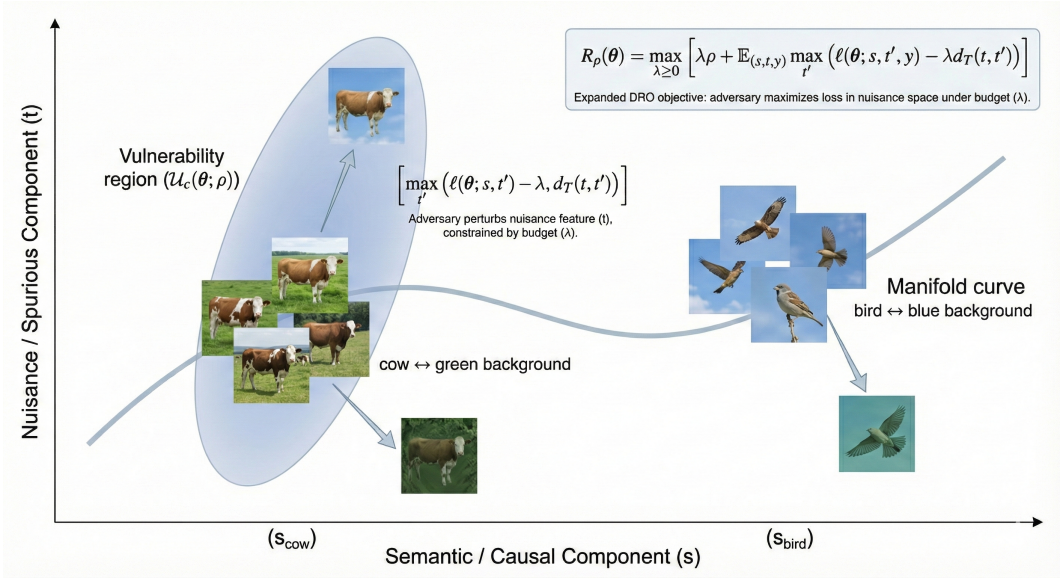


Figure 1: Illustration of our viewpoint of domain generalization through a distributionally robust optimization lens. The horizontal axis represents the semantic/causal component s (e.g., *cow* vs. *bird*), while the vertical axis captures nuisance or spurious factors t (e.g., background color). An adversary moves along the data manifold by perturbing only t, keeping the semantic concept and label (s, y) fixed under a nuisance-only transport budget, and thus searches within the vulnerability region $U_c(\theta; \rho)$ for nuisance variations that maximally increase the loss, leading to the expanded DRO objective $\mathcal{R}_\rho(\theta)$. This perspective motivates learning predictors that remain stable under worst-case nuisance shifts while relying on invariant, causal structure.

1 Introduction

A classifier trained to recognize cows on a biased dataset of pastoral scenes often learns to associate the label with grass texture rather than with cow shape [1]. Its failure on a cow photographed at a beach is a predictable consequence of Empirical Risk Minimization (ERM). On the source data, a gradient direction that improves a cow-shaped detector and a gradient direction that improves a grass-based detector can both reduce the training loss. ERM receives no reason to prefer the former when the shortcut is equally predictive in training [2–4]. The question we address is which component of a per-sample ERM update is responsible for shortcut behavior, and how to remove that component without losing the rest of the learning signal.

Domain Generalization (DG) has produced a wide range of methods to address this failure mode. Data augmentation expands the training distribution through image-space or feature-space transformations such as Mixup [8] and style mixing [9]. Invariance-based methods align feature statistics or predictors across source domains, including moment matching [10], domain-adversarial training [11, 12], and the invariant risk principle [4, 13, 14]. Learning-strategy methods modify the optimization itself, for example through meta-learning [15] or by aligning gradient statistics computed on different domains [16–18]. These approaches differ in mechanism but share a structural assumption; the generalization signal is constructed by comparing two or more source environments. The assumption breaks down in three regimes that motivate this paper. When the same spurious cue is present across all source domains, the inter-domain signal can preserve the shortcut. When domain labels are unavailable, the comparison cannot be performed. When several inter-domain alignment terms are combined, the resulting loss surface can become sensitive to dataset-specific hyperparameter tuning. Recent DG methods also add external priors or task-specific machinery, including MIRO [19], logical reasoning regularization [20], causal semantic-consistency reasoning [21], flow-factorized state-space DG architectures [22], and cross-modal Schrödinger-bridge DG [23]. CGS is complementary: it is an optimizer-side, domain-label-free correction whose additional signal comes from a destroyed view of the same minibatch rather than from domain-wise statistics, explicit logical formulas, cross-modal generation, or a specialized backbone.

This paper takes a different starting point. We ask what survives an intervention applied to a single training sample. The intervention is a destruction operator that suppresses the semantic content of an input while preserving its nuisance content. Applied to a cow-on-grass image, the operator might shuffle local patches so that cow shape is broken while grass texture is left intact. If the model can still predict the label from this destroyed view, the gradient of that prediction points along a direction in parameter space the model is using without semantic justification. The destroyed view is a counterfactual probe derived from the same training sample as the clean view. In implementation, these clean and destroyed losses are averaged over the minibatch for efficiency before the layerwise projection is applied; this is a standard stochastic approximation to the samplewise contrast, not a cross-domain comparison. The resulting correction signal differs structurally from cross-sample negative mining and contrastive selection mechanisms, which rely on separate samples with comparable or opposing features [24, 25], and from causal-representation methods that aim to recover latent causal variables [26–31, 21]. Our claim is optimizer-facing. The destroyed view is used only to estimate a direction in parameter space; the supervision for descent remains the original sample. For near-ERM efficiency the practical probe is invoked every k iterations, while intervening iterations follow the base optimizer.

The same principle admits a Distributionally Robust Optimization (DRO) interpretation. A nuisance-only adversary, allowed to transport mass within nuisance space while keeping semantics fixed, defines a Wasserstein ambiguity set whose dual form is a per-sample inner maximization over nuisance perturbations [32]. The local Danskin majorizer of this dual induces a first-order robust-feasible cone in parameter space whose normal is the nuisance-certificate gradient. Causal Gradient Steering (CGS) is the closed-form Euclidean projection of the ERM step onto a tractable single-normal approximation of this cone. We further prove that an approximately admissible destruction probe, meaning a probe whose output is close to the ideal nuisance-only view in optimizer-relevant coordinates, is sufficient to recover the cone normal up to controlled error, preserve the gating decision under a margin, and keep the implemented update close to the ideal projected update. Exact counterfactual generation in pixel space is unnecessary. We also establish a target-risk bound for causally compatible nuisance shifts, in which target loss is controlled by empirical nuisance-DRO risk, statistical complexity, semantic shift, and accumulated cone-violation and probe-admissibility surcharges from training.

We view CGS as the first instance of a broader principle. Data augmentation expanded the training distribution by creating new training views of existing samples, and gave rise to a family of techniques organized by the design of the transformation [33, 8, 34–36]. Causal signal destruction defines a parallel design space with a different role for the transformed view; it is a probe used to estimate which directions in parameter space the model treats as predictive without semantic justification. The choice of destruction operator encodes a structural prior about what is causal and what is nuisance for a given task. In classification, local patch shuffling can suppress object-level shape while preserving texture and background statistics. In tracking, a destroyer can preserve the template while disrupting target evidence inside the search tube. In semantic segmentation, local center-patch destruction can expose shortcut gradients while leaving the dense-prediction training pipeline unchanged. More generally, each vision task may admit its own family of destroyers, and CGS provides a way to turn such operators into optimization-time shortcut corrections.

Contributions. This paper makes the following contributions.

- We introduce *causal signal destruction*, an anti-augmentation principle for robust generalization. Instead of treating transformed samples as additional label-preserving data, we deliberately destroy task-defining evidence and use the remaining predictive gradient as a probe for shortcut learning.
- We instantiate this principle as *Causal Gradient Steering* (CGS), a domain-label-free optimizer that contrasts clean and destroyed gradients within the same minibatch and removes nuisance-aligned components of the ERM update.
- We derive CGS from a nuisance-only causal DRO geometry. We show that CGS is the closed-form projection of the ERM step onto a tractable single-normal approximation of a first-order robust-feasible cone in parameter space induced by the local majorizer. We further prove target-risk and probe-admissibility guarantees, showing that exact pixel-space counterfactual generation is unnecessary.
- CGS obtains the strongest average among the listed DG benchmark results while retaining near-ERM amortized training and unchanged inference cost. Under one locked configuration,

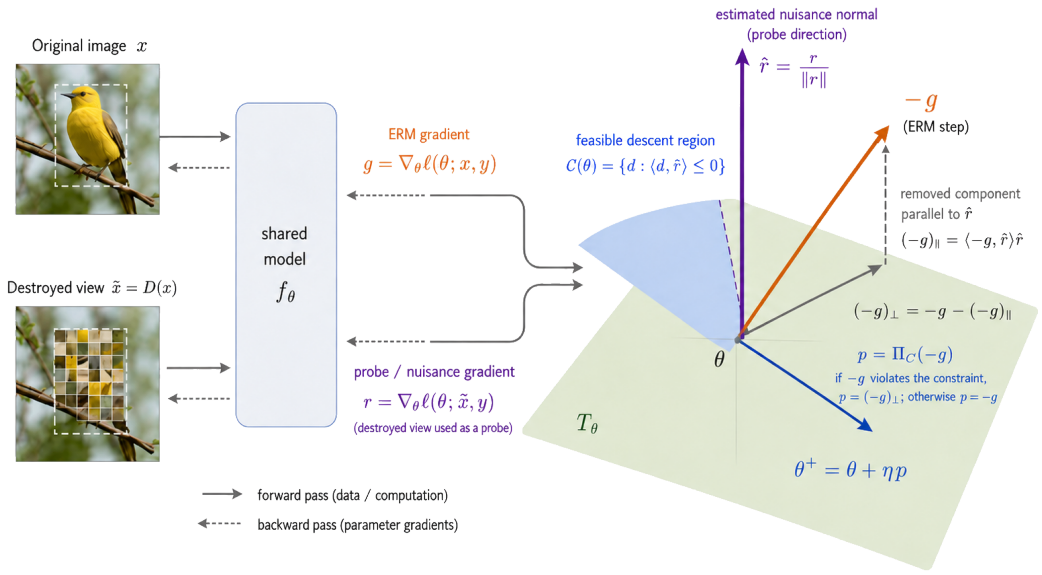


Figure 2: Overview of Causal Gradient Steering (CGS). The clean view x and destroyed view $\tilde{x} = \mathcal{D}(x)$ are passed through the same model. The clean loss produces the ERM gradient g, while the destroyed view is used only as a probe to estimate a nuisance-aligned normal r. CGS normalizes this direction to $\hat{r}$ and projects the descent step $-g$ onto the feasible cone $\mathcal{C}(\theta) = \{d : \langle d, \hat{r} \rangle \leq 0\}$, removing the component that would increase nuisance fit. In the implementation, this projection is applied layerwise and periodically using minibatch-averaged gradients.

it improves seventeen DG methods by up to +15.8 points and transfers to tracking and domain-generalized semantic segmentation.

2 Methodology

Object shape and surrounding context may both predict the training label, and the ERM gradient rewards both. CGS asks what remains predictive when the structure that gives an image its class identity is disrupted. We investigate this by computing a second gradient on a deliberately destroyed view of the same sample. Rather than learning from the damaged image, we use its gradient to probe directions that may still be supported by nuisance information. The challenge is then to use this signal without sacrificing useful learning from the original image. We approach this through robust optimization, which provides the basis for the practical gradient correction developed in the following subsections.

2.1 Setup and Destruction Operator

We use the causal decomposition standard in work on invariance and generalization [26, 27]. The observed input x is written as (s, t), where $s \in \mathcal{S}$ is the semantic component that determines the label and $t \in \mathcal{T}$ is a nuisance component that can become spuriously correlated with the label. In the cow example, s contains the cow shape and class-defining content, while t contains background, texture, style, and acquisition factors such as grass. The structural causal model used by the theory is shown in Figure 3.

ERM minimizes

$$\min_\theta L_{\mathrm{ERM}}(\theta) = \mathbb{E}_{(s,t,y)\sim P}\big[\ell(\theta; s, t, y)\big]. \tag{1}$$

For a mini-batch ν, write

$$\Delta R_\nu(\theta) = \frac{1}{|\nu|} \sum_{(x_i,y_i)\in\nu} \ell(\theta; x_i, y_i), \qquad g = \nabla_\theta \Delta R_\nu(\theta).$$

The gradient g can mix semantic and nuisance-sensitive directions, since both may reduce the source loss.

Suppose the semantic evidence could be removed while leaving nuisance factors untouched. Any remaining label evidence would then come from those factors. We write this ideal operation as

$$\mathcal{D}(s,t) = (s_\varnothing, t), \tag{2}$$

where $s_\varnothing$ is a null semantic state. Equation (2) is a theoretical ideal: practical destruction operators instead act on observed inputs, aiming to weaken semantic evidence while retaining nuisance cues without explicitly separating s and t. To measure label fit on the ideal nuisance-only view, define

$$J(\theta) = \sum_i \log q_\theta(y_i \mid s_\varnothing, t_i), \tag{3}$$

whose ascent direction is

$$r = \nabla_\theta J(\theta). \tag{4}$$

The normal r identifies parameter changes that strengthen prediction from nuisance evidence alone. Consider a cow on grass. The clean loss gradient may reward both the animal's shape and the surrounding grass, while a gradient from a view with disrupted cow structure probes what remains predictive from the grass-like cues. When the clean and destroyed loss gradients align, part of the clean update may also improve predictions without the intact cow. CGS uses this overlap as a shortcut-sensitive signal while keeping supervised descent on the clean image.

Standing notation. Throughout this section, g denotes the clean ERM gradient, r the ideal nuisance-certificate normal, $u = r/\|r\|_2$ its unit normalization when $r \neq 0$, $\tilde{r}$ a probe-estimated normal, and $\hat{u} = \tilde{r}/\|\tilde{r}\|_2$. For a candidate update $\Delta\theta$, cone feasibility means that the first-order nuisance certificate does not increase.

2.2 Nuisance-Only DRO Geometry

We consider domain shifts that change nuisance factors while leaving the task semantics intact. To model this uncertainty, we introduce a transport cost that allows changes in nuisance space but assigns infinite cost to changes in the semantic component:

$$c\big((s,t),(s',t')\big) = \begin{cases} d_T(t,t') & \text{if } s = s', \\ +\infty & \text{if } s \neq s', \end{cases} \tag{5}$$

where d_T is a metric on nuisance space. The associated Wasserstein ambiguity set gives the nuisance-only distributionally robust optimization (DRO) risk,

$$\mathcal{R}_\rho(\theta) = \sup_{Q: W_c(Q,P)\leq\rho} \mathbb{E}_{(s,t,y)\sim Q}\big[\ell(\theta; s, t, y)\big]. \tag{6}$$

Following the Wasserstein dual formulation, this risk can be expressed as a collection of pointwise nuisance searches [32]:

$$\mathcal{R}_\rho(\theta) = \min_{\lambda\geq 0} \left\{ \lambda\rho + \frac{1}{n}\sum_{i=1}^{n} \sup_{t'\in\mathcal{T}} \big(\ell(\theta; s_i, t', y_i) - \lambda d_T(t', t_i)\big) \right\}. \tag{7}$$

One might view the inner maximization in Eq. (7) as a hypothetical adversary searching for nuisance configurations under which the model performs poorly. For each training example, it keeps the semantic component and label fixed while considering alternative nuisance factors, such as background, texture, or style. Changes that move farther from the observed nuisance configuration incur a larger transport penalty, controlled by λ, while the outer minimization balances this penalty against the transport budget ρ. This adversarial search is part of the robustness model rather than an optimization procedure carried out by CGS.

The formulation tells us what robustness means, but not yet how to achieve it through gradient updates. A parameter change that reduces the training loss may still increase the loss under unfavorable nuisance conditions. Moreover, the nuisance configuration that maximizes the loss can change as the model changes. To understand this behavior, we study how the robust risk varies locally with the model parameters, accounting for the gradients associated with its maximizing nuisance configurations.

These configurations are generally unavailable. For the theoretical analysis, we therefore introduce a class-conditional *nuisance bank* $\mathcal{B}_c$, consisting of idealized destroyed views paired with their latent nuisance factors. The bank provides reference nuisance configurations, and a coverage assumption requires that the configurations relevant to the inner maximization lie sufficiently close to these references.

For a candidate parameter update $\Delta\theta$, we define the bank-supported nuisance sensitivity as

$$\Gamma_\theta(\Delta\theta) = \frac{1}{n}\sum_{i=1}^{n} \sup_{(\tau,t_\tau)\in\mathcal{B}_{y_i}} \Big\langle \nabla_\theta \ell(\theta; s_i, t_\tau, y_i) - \nabla_\theta \ell(\theta; s_i, t_i, y_i),\, \Delta\theta \Big\rangle.$$

This quantity compares the first-order effect of an update under alternative nuisance conditions with its effect on the observed training examples. Taking the supremum identifies the largest such difference supported by the bank. A positive value indicates that the update can increase vulnerability to nuisance variation beyond what is captured by the ordinary training gradient.

The following result uses a local Danskin argument to bound this additional vulnerability.

Theorem 2.1 (Local Danskin DRO Majorizer)**.** Under the regularity, local upper-model, and bank-coverage assumptions in Appendix B, every sufficiently small perturbation $\Delta\theta$ satisfies

$$\mathcal{R}_\rho(\theta+\Delta\theta) \le \mathcal{R}_\rho(\theta) + \langle g, \Delta\theta\rangle + [\Gamma_\theta(\Delta\theta)]_+ + M\varepsilon_{\mathrm{cov}}\|\Delta\theta\| + \frac{\beta}{2}\|\Delta\theta\|^2 + \omega_\theta(\|\Delta\theta\|)\|\Delta\theta\|, \quad (8)$$

where $\varepsilon_{\mathrm{cov}}$ measures bank-coverage error, M and β are smoothness constants, and $\omega_\theta(r) \to 0$ as $r \downarrow 0$ accounts for local changes in the maximizing nuisance configurations. The induced first-order cone is

$$\mathcal{C}_\theta^{\mathrm{bank}} = \big\{\Delta\theta : \Gamma_\theta(\Delta\theta) \le 0\big\}.$$

The bound separates the usual ERM contribution from the additional first-order sensitivity to nuisance variation. When an update belongs to $\mathcal{C}_\theta^{\mathrm{bank}}$, the positive nuisance surcharge vanishes, leaving only the stated coverage and higher-order terms. Enforcing this constraint directly, however, would require access to the nuisance bank and its relevant configurations. In the next subsection, we develop a tractable approximation using a single direction estimated from the destroyed view.

2.3 CGS as Projection

Projecting onto the full bank cone would require the active nuisance candidates. CGS instead uses the gradient r of J to construct a single-normal approximation: the half-space below contains updates that do not increase nuisance-only label fit to first order.

Definition 2.2 (Single-Normal Cone)**.** Let $r = \nabla_\theta J(\theta)$. The single-normal local robust cone is

$$\mathcal{C}_\theta = \{\Delta\theta : \langle r, \Delta\theta\rangle \le 0\}.$$

Proposition 2.3 (First-Order Nuisance Feasibility)**.** If J is differentiable at θ, then

$$J(\theta+\Delta\theta) = J(\theta) + \langle r, \Delta\theta\rangle + o(\|\Delta\theta\|).$$

Hence updates in $\mathcal{C}_\theta$ do not increase J to first order, while $\langle r, \Delta\theta\rangle > 0$ increases the nuisance certificate to first order.

This half-space is the tractable approximation used by the algorithm. We do not assume that it equals $\mathcal{C}_\theta^{\mathrm{bank}}$ in full generality. It compresses the exact support-function geometry into one normal that can be estimated by a probe. Appendix F gives a sufficient domination condition under which a margin-shrunken single-normal cone is conservative for the bank cone; outside that condition, the single-normal cone is the deliberate algorithmic compression used for efficiency.

ERM proposes the raw step $-\eta g$. CGS chooses the closest step that lies in $\mathcal{C}_\theta$,

$$\Delta\theta^\star = \arg\min_{\Delta\theta} \frac{1}{2}\|\Delta\theta - (-\eta g)\|^2 \quad \text{s.t.} \quad \langle r, \Delta\theta\rangle \le 0. \quad (9)$$

Theorem 2.4 (Exact Single-Normal Projection)**.** Let $\hat{r} = r/\|r\|_2$. The unique solution of Eq. (9) is

$$\Delta\theta^\star = \begin{cases} -\eta g, & \text{if } \langle g, \hat{r}\rangle \geq 0, \\ -\eta\big(g - \langle g, \hat{r}\rangle \hat{r}\big), & \text{if } \langle g, \hat{r}\rangle < 0. \end{cases} \tag{10}$$

If $\langle g, \hat{r}\rangle \geq 0$, the ERM step already lies in the cone and is left unchanged. If $\langle g, \hat{r}\rangle < 0$, the ERM step would increase nuisance fit to first order; CGS removes exactly that component and returns the nearest feasible step on the cone boundary. The projection is therefore the optimizer of the tractable cone subproblem induced by the nuisance-only DRO argument, giving the gradient correction a direct robust-optimization interpretation.

2.4 Soft and Gated CGS

The hard projection enforces cone feasibility exactly. A soft variant penalizes only cone violation,

$$\min_{\Delta\theta} \frac{1}{2\eta}\|\Delta\theta + \eta g\|_2^2 + \frac{\mu}{2}[\langle \Delta\theta, \hat{u}\rangle]_+^2. \tag{11}$$

The penalty is zero for cone-feasible motion and grows only in the nuisance-increasing half-space.

Proposition 2.5 (Soft/Gated Relaxation)**.** Let $a = \langle g, \hat{u}\rangle$, $I_- = \mathbb{I}\{a < 0\}$, and $\gamma = \eta\mu/(1 + \eta\mu) \in [0, 1)$. The unique minimizer of Eq. (11) is

$$\Delta\theta^\star = -\eta\big(g - \gamma I_- a\hat{u}\big). \tag{12}$$

The correction is applied only when the ERM step is cone-infeasible, and then shrinks the nuisance-increasing component by strength γ.

Gating follows from the one-sided penalty. Cone-feasible directions receive no penalty, while directions that would increase the nuisance certificate are pulled back toward the cone. The limit $\gamma \uparrow 1$ recovers the hard projection; Appendix H gives the derivation. All proximal statements use $\gamma \in [0, 1)$; the default reported configuration uses $\gamma = 0.5$. Sensitivity values with $\gamma > 1$ in Appendix U are explicitly treated as over-relaxed engineering variants and are not invoked by the proximal proof.

2.5 Practical Probes and Blockwise Implementation

The nuisance component is unobserved, so we estimate its normal r with a probe $\mathcal{P}(x)$ (Figure 2). For images, patch shuffling breaks object structure while retaining local textures, colors, and styles. The gradient of this view guides the correction to the clean update; the destroyed view does not become another training example.

Selecting a probe. The destruction scale depends on the task: patch shuffling disrupts object layout in classification, finer tiles break continuity in segmentation, and tracking disrupts search-region evidence while preserving the template. The held-out probes in Appendix U.2 examine what class and domain information remains after shuffling. CGS training needs no domain or shortcut annotations.

Approximate admissibility makes this requirement precise. In Appendix J, a probe is $(\varepsilon_s, \varepsilon_t)$-admissible when its representation is close to the ideal nuisance-only view, within semantic-leakage and nuisance-distortion tolerances. The bounds there control the estimated normal, gate stability, and the gap between practical and ideal updates.

Proposition 2.6 (Positive Alignment is Sufficient)**.** Let $\hat{u} = \tilde{r}/\|\tilde{r}\|_2$, $a = \langle g, \hat{u}\rangle$, and $I_- = \mathbb{I}\{a < 0\}$. For

$$\Delta\theta_{\mathrm{ERM}} = -\eta g, \qquad \Delta\theta_{\mathrm{CGS}} = -\eta\big(g - \gamma I_- a\hat{u}\big),$$

the true nuisance-certificate change relative to ERM satisfies

$$\langle r, \Delta\theta_{\mathrm{CGS}} - \Delta\theta_{\mathrm{ERM}}\rangle = \eta\gamma I_- a\langle r, \hat{u}\rangle. \tag{13}$$

Thus, when $\langle r, \hat{u}\rangle > 0$ and the gate is active, CGS decreases the true nuisance certificate more than ERM does to first order.

Algorithm 1: Practical Causal Gradient Steering

Require: model f_θ with layers $\{\theta_l\}_{l=1}^L$, destroyer $\mathcal{P}$, loss ℓ, optimizer
Require: strength $\gamma \geq 0$ (proximal range $[0, 1]$; > 1 only as over-relaxation), probe period k, dot-score threshold $\tau_{\mathrm{dp}} \geq 0$, numerical constant ϵ
for training step $t = 1, 2, \ldots$ **do**
  sample minibatch $B = \{(x_i, y_i)\}_{i=1}^b$
  $\ell_B(\theta) \leftarrow b^{-1} \sum_i \ell(f_\theta(x_i), y_i)$
  $g_l \leftarrow \nabla_{\theta_l} \ell_B(\theta)$ for all layers l
  if $t \bmod k = 0$ **then**
    $\ell_{\mathcal{P}(B)}(\theta) \leftarrow b^{-1} \sum_i \ell(f_\theta(\mathcal{P}(x_i)), y_i)$
    $\tilde{h}_l \leftarrow \nabla_{\theta_l} \ell_{\mathcal{P}(B)}(\theta)$ for all layers l
    for layer $l = 1, \ldots, L$ **do**
      $s_l \leftarrow \langle g_l, \tilde{h}_l \rangle; \quad \rho_l \leftarrow s_l / (\|g_l\| \, \|\tilde{h}_l\| + \epsilon)$
      if $s_l > \tau_{\mathrm{dp}}$ **then**
        $\alpha_l \leftarrow \gamma s_l / (\|\tilde{h}_l\|^2 + \epsilon)$
        $\hat{g}_l \leftarrow g_l - \alpha_l \tilde{h}_l$
        $g_l \leftarrow \|g_l\| \hat{g}_l / (\|\hat{g}_l\| + \epsilon)$
  update θ using gradients $\{g_l\}_{l=1}^L$

Corollary 2.7 (Admissibility Yields Alignment and Gate Stability)**.** Let $\delta_r = L_s \varepsilon_s + L_t \varepsilon_t$. If $\delta_r < \|r\|_2$, then

$$\|\hat{u} - u\|_2 \leq \frac{2\delta_r}{\|r\|_2}, \qquad \langle \hat{u}, u \rangle \geq 1 - 2\frac{\delta_r^2}{\|r\|_2^2}.$$

In particular, the alignment is strictly positive whenever $\delta_r < \|r\|_2/\sqrt{2}$. If the ideal margin obeys

$$\langle g, u \rangle \leq -\Big(\tau + \frac{2\delta_r}{\|r\|_2}\Big)\|g\|_2,$$

then the practical gate also fires, $\langle g, \hat{u} \rangle \leq -\tau \|g\|_2$.

Modern networks are optimized over parameter blocks, so CGS is applied layerwise. With $\theta = (\theta_1, \ldots, \theta_L)$, ERM block gradients g_l, and unit estimated normals $\hat{u}_l$, the block-separable relaxation is

$$\min_{\{\Delta\theta_l\}_{l=1}^L} \sum_{l=1}^L \left[\frac{1}{2\eta_l} \|\Delta\theta_l + \eta_l g_l\|_2^2 + \frac{\mu_l}{2} [\langle \Delta\theta_l, \hat{u}_l \rangle]_+^2 \right]. \tag{14}$$

Proposition 2.8 (Blockwise Margin-Gated Cone Update)**.** For each layer l, let $a_l = \langle g_l, \hat{u}_l \rangle$, $I_-^{(l)} = \mathbb{I}\{a_l < 0\}$, and $\gamma_l = \eta_l \mu_l / (1 + \eta_l \mu_l)$. Eq. (14) decomposes across layers and has optimizer

$$\Delta\theta_l^\star = -\eta_l \big(g_l - \gamma_l I_-^{(l)} a_l \hat{u}_l\big), \qquad l = 1, \ldots, L.$$

A cosine-style margin gate replaces $a_l < 0$ by $a_l < -\tau_{\cos}\|g_l\|_2$ to avoid weak or noisy nuisance certificates. The reported code uses the equivalent destroyed-gradient sign logic with a small raw dot-score threshold for numerical sparsity; Appendix I gives the sign-convention details. The probe period k is the number of optimization steps between destroyed-view probes (Algorithm 1).

Algorithm 1 is written in optimizer-gradient form; Appendix I explains the destroyed-view sign convention, raw dot-score gate, and the diagnostic cosine score ρ_l. The norm-rescaling line preserves the original layerwise step budget while changing only the update direction; Appendix M gives the one-step decrease bound for the rescaled update.

2.6 Theory Summary and Appendix Map

The local geometry above also yields a target-risk guarantee for causally compatible nuisance shifts. In the formal result, the target may change $T \mid S, Y$ while preserving the semantic-label mechanism, and the target risk is bounded by five terms: empirical nuisance-DRO risk, statistical complexity of the restricted DRO class, semantic-label shift, accumulated cone violation, and probe-admissibility surcharge. The full statement is Theorem L.1 in Appendix L.

This bound makes the role of the cone explicit. If updates stay inside the exact bank-induced cone, the first-order cone-violation surcharge vanishes. With an approximate probe, Corollary L.2 shows that the target-risk inflation from probe error is linear in $L_s \varepsilon_s + L_t \varepsilon_t$, scaled by the update norm and the strength of the true nuisance normal. The full proof chain, including the empirical DRO concentration step and the telescoping argument over training, is in Appendix L.

Table 1: Average classification accuracy (%) on standard DG benchmarks under the leave-one-domain-out protocol. Prior-method values are cited published or established benchmark results; CGS is run under the stated DomainBed-style setup. Best results are in **bold**. Uncertainty values are source-reported when available; CGS values are computed from 3 repeated runs under the official DomainBed protocol.

Model	Venue	PACS	VLCS	OfficeHome	TerraIncognita	DomainNet	Avg
ERM	Baseline	85.5 ± 0.2	77.5 ± 0.4	66.5 ± 0.3	46.1 ± 1.8	41.3 ± 0.1	63.4
IRM [4]	ArXiv '19	83.5 ± 0.8	78.6 ± 0.5	64.3 ± 2.2	47.6 ± 0.8	28.0 ± 2.8	60.4
GDRO [41]	ICLR '20	84.4 ± 0.8	76.7 ± 0.6	66.0 ± 0.7	43.2 ± 1.1	33.4 ± 0.2	60.7
Mixup [8]	ICLR '18	84.6 ± 0.6	77.4 ± 0.6	68.1 ± 0.3	47.9 ± 0.8	39.6 ± 0.1	63.5
MLDG [15]	AAAI '18	84.9 ± 1.0	77.2 ± 0.4	66.8 ± 0.6	47.7 ± 0.9	41.6 ± 0.1	63.6
CORAL [10]	ECCV '16	86.2 ± 0.3	78.8 ± 0.6	68.7 ± 0.3	47.6 ± 1.0	41.8 ± 0.1	64.6
MMD [12]	CVPR '18	84.6 ± 0.5	77.5 ± 0.9	66.3 ± 0.1	42.2 ± 1.6	23.5 ± 9.5	58.8
DANN [11]	JMLR '16	83.6 ± 0.4	78.6 ± 0.4	65.9 ± 0.6	46.7 ± 0.5	38.3 ± 0.1	62.6
CDANN [42]	ECCV '18	82.6 ± 0.9	77.5 ± 0.1	65.8 ± 1.3	45.8 ± 1.6	38.5 ± 0.3	62.0
MTL [43]	JMLR '21	84.6 ± 0.5	77.2 ± 0.4	66.4 ± 0.5	45.6 ± 1.2	40.6 ± 0.1	62.9
SagNet [44]	CVPR '20	86.3 ± 0.2	77.8 ± 0.5	68.1 ± 0.1	48.6 ± 1.0	40.3 ± 0.1	64.3
MixStyle [9]	ICLR '21	85.2	77.9	60.4	44.0	34.0	60.3
ARM [45]	NeurIPS '21	85.1 ± 0.4	77.6 ± 0.3	64.8 ± 0.3	45.5 ± 0.3	36.0 ± 0.2	61.8
VREx [14]	ICML '21	84.9 ± 0.6	78.3 ± 0.2	66.4 ± 0.6	46.4 ± 0.6	30.1 ± 2.9	61.2
RSC [46]	ECCV '20	85.2 ± 0.9	77.1 ± 0.5	65.5 ± 0.9	46.6 ± 1.0	38.9 ± 0.5	62.7
mDSDI [47]	NeurIPS '21	86.2 ± 0.2	79.0 ± 0.3	69.2 ± 0.4	48.1 ± 1.4	42.8 ± 0.1	65.1
SWAD [48]	NeurIPS '21	88.1 ± 0.1	79.1 ± 0.1	70.6 ± 0.2	50.0 ± 0.3	46.5 ± 0.1	66.9
SMA [49]	NeurIPS '22	87.5 ± 0.2	78.2 ± 0.2	70.6 ± 0.1	50.3 ± 0.5	46 ± 0.1	66.5
Fishr [18]	ICML '22	86.9 ± 0.2	78.2 ± 0.2	68.2 ± 0.2	**53.6** ± 0.4	41.8 ± 0.2	65.7
PCL [50]	CVPR '22	88.7	-	71.6	52.1	47.7	65.0
MIRO [19]	ECCV '22	85.4 ± 0.4	79.0 ± 0.0	70.5± 0.4	50.4 ± 1.1	44.3 ± 0.2	65.9
SAGM [51]	CVPR '23	86.6 ± 0.2	**80.0** ± 0.3	70.1 ± 0.2	48.8 ± 0.9	45.0 ± 0.2	66.1
INSURE [52]	TIP '24	**89.2** ± 0.3	79.1	72.0 ± 0.1	53.1 ± 0.5	48.0 ± 0.1	68.3
Arith [53]	CVPR '25	86.5 ± 0.3	79.4 ± 0.3	69.4 ± 0.1	48.1 ± 1.2	41.5 ± 0.1	65.0
GGA [54]	CVPR '25	87.3 ± 0.4	79.9 ± 0.4	68.5 ± 0.2	50.6 ± 0.1	45.2 ± 0.2	66.3
CGS (Ours)	-	89.1 ± 0.2	79.5 ± 0.1	**73.6** ± 0.1	52.7 ± 0.1	**50.1** ± 0.1	**69.0**

Table 2: OOD generalization results on WILD's FMOW and PCAM.

Method	FMOW	PCAM
ERM	46.6	91.8
ERM++ w/ ResNet-50 Init. [55]	48.1	89.4
CGS (Ours)	**55.3**	**92.9**

3 Experiments

We evaluate CGS along three axes: standard DomainBed-style classification benchmarks (PACS, VLCS, OfficeHome, TerraIncognita, DomainNet), WILDS FMoW [38, 39] and PCAM [40, 39], together with controlled shortcut benchmarks, plug-in correction of existing DG algorithms under one locked CGS configuration, and transfer to tracking and domain-generalized semantic segmentation. Table 1 is a reference benchmark comparison using published or established prior-method numbers under the stated protocol; wrapper and structured-task comparisons use matched base-pipeline settings. Full implementation details, per-domain breakdowns, plug-in results, single-source DG results, compute comparisons, qualitative analyses, ablations, and structured-task tables are in the appendix.

3.1 Results Overview

Table 1 shows that CGS is competitive with, and often stronger than, leading DG baselines spanning invariance, augmentation, alignment, and gradient-based families. Relative to ERM, CGS improves the five-dataset average from 63.4% to 69.0%, with the strongest reported averages on OfficeHome (73.6%) and DomainNet (50.1%). These gains are consistent with the core mechanism: a destruction probe estimates a nuisance-aligned direction, and CGS removes that component from the clean ERM update.

Controlled shortcut benchmarks let us test the idea where spurious correlations are explicit. CGS improves worst-group accuracy over matched ERM by 12.3, 27.0, and 5.9 points on ColoredMNIST, Waterbirds, and CelebA, respectively (Appendix U.3). We probe background dependence more directly by keeping 130 OfficeHome foreground objects fixed while replacing their scenes. Both models begin at 96.9% accuracy; CGS loses 17.9 points after replacement versus 21.3 for ERM, with fewer prediction flips (18.5% versus 22.0%; Appendix U.4).

The ablations then test the steering rule itself. All six modifications reduce accuracy on TerraIncognita and PACS; probing every step has the largest cost (4.56 and 3.25 points), supporting the periodic update (Appendix U.1).

Beyond classification, Table 3 summarizes structured-task transfer. With the same locked CGS hyperparameter configuration as the plug-in DG experiments, CGS improves OSTrack-256 across tracking benchmarks and improves matched-protocol SCSD averages for domain-generalized semantic segmentation. Full breakdowns and analyses are provided in the appendix. Matched class-level comparisons show mean IoU gains for several small foreground categories, including traffic lights (+0.79), traffic signs (+1.23), bicycles (+10.88), and motorcycles (+2.59) across six target conditions (Appendix T.1).

Table 3: Compact transfer results beyond classification using the same locked CGS hyperparameter configuration; full tables are in Appendices S and T.

Single object tracking

Method	GOT AO	UAV AUC	OTB AUC
OSTrack-256	71.0	65.79	66.68
+ CGS	**72.4**	**66.14**	**68.37**

Domain-generalized segmentation

Setting	SCSD	+ CGS	Gain
G+S→C/B/M	51.42	**52.35**	+0.93
G→ACDC	35.66	**36.99**	+1.33

Across these experiments, the destruction operator changes with the task, but the optimizer-side role remains the same: damage task-defining evidence to expose shortcut gradients, then remove their aligned component from the clean update. The destroyed branch is never optimized as a label-preserving augmentation; this design choice eliminates the most direct augmentation confound. The appendix further probes the mechanism through destroyer-family sensitivity, intervention visualizations, loss-landscape slices, and Grad-CAM comparisons.

4 Conclusion

We introduced *Causal Gradient Steering* (CGS), a domain-label-free optimizer grounded in nuisance-only causal DRO. The theory connects the local Danskin majorizer to a first-order robust-feasible cone and turns this geometry into a tractable single-normal projection estimated by an admissible destruction probe. Empirically, CGS obtains the strongest average among the listed five-benchmark DG results, improves seventeen existing DG algorithms by up to +15.8 points under one locked configuration, and transfers to tracking and semantic segmentation through task-specific destroyers. CGS therefore offers a practical framework for robust generalization by steering away from nuisance-aligned gradients; stronger semantic selection bias and broader modalities remain future work.

Acknowledgments and Disclosure of Funding

Funding. This research was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) through a Discovery Grant (RGPIN-2022-03015). Computational resources were provided by Advanced Research Computing at the University of British Columbia (UBC ARC Sockeye) and Compute Canada (now the Digital Research Alliance of Canada).

Competing interests. The authors declare no competing interests.

References

[1] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 456–473, 2018.

[2] Joaquin Quiñonero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D Lawrence. *Dataset shift in machine learning*. MIT press, 2009.

[3] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8340–8349, 2021.

[4] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[5] 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*, 2:665–673, 2020. doi: 10.1038/s42256-020-00257-z.

[6] 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 (ICLR)*, 2020.

[7] 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.

[8] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations (ICLR)*, 2018.

[9] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=6xHJ37MVxxp`.

[10] Baochen Sun and Kate Saenko. Deep CORAL: correlation alignment for deep domain adaptation. *CoRR*, abs/1607.01719, 2016. URL `http://arxiv.org/abs/1607.01719`.

[11] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research (JMLR)*, 17(59):1–35, 2016.

[12] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5400–5409, 2018.

[13] Kartik Ahuja, Karthikeyan Shanmugam, Kush Varshney, and Amit Dhurandhar. Invariant risk minimization games. In *International Conference on Machine Learning*, pages 145–155. PMLR, 2020.

[14] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Rémi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (vrex). In *International Conference on Machine Learning (ICML)*, pages 5815–5826. PMLR, 2021.

[15] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[16] Yuge Shi, Jeffrey Seely, Philip Torr, Siddharth N, Awni Hannun, Nicolas Usunier, and Gabriel Synnaeve. Gradient matching for domain generalization. 2022. URL `https://openreview.net/forum?id=vDwBW49HmO`.

[17] Lucas Mansilla, Rodrigo Echeveste, Diego H Milone, and Enzo Ferrante. Domain generalization via gradient surgery. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6630–6638, 2021.

[18] Alexandre Rame, Corentin Dancette, and Matthieu Cord. Fishr: Invariant gradient variances for out-of-distribution generalization. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 18347–18377. PMLR, 17–23 Jul 2022. URL `https://proceedings.mlr.press/v162/rame22a.html`.

[19] Junbum Cha, Kyungjae Lee, Sungrae Park, and Sanghyuk Chun. Domain generalization by mutual-information regularization with pre-trained models. In *European conference on computer vision*, pages 440–457. Springer, 2022.

[20] Zhaorui Tan, Xi Yang, Qiufeng Wang, Anh Nguyen, and Kaizhu Huang. Interpret your decision: Logical reasoning regularization for generalization in visual classification. *Advances in Neural Information Processing Systems*, 37:18166–18204, 2024.

[21] Zongxin Liu, Yishu Liu, Guangming Lu, Xiaoling Luo, and Bingzhi Chen. Caurdg: Enhancing domain generalization with causal-driven semantic consistency reasoning. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 11101–11110, 2025.

[22] Qi Bi, Jingjun Yi, Hao Zheng, Haolan Zhan, Wei Ji, Yawen Huang, and Yuexiang Li. Dgfamba: Learning flow factorized state space for visual domain generalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 1862–1870, 2025.

[23] Hao Zheng, Jingjun Yi, Qi Bi, Huimin Huang, Haolan Zhan, Yawen Huang, Yuexiang Li, Xian Wu, and Yefeng Zheng. Learning a cross-modal schrödinger bridge for visual domain generalization. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen, editors, *Advances in Neural Information Processing Systems*, volume 38, pages 167613–167641. Curran Associates, Inc., 2025.

[24] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning (ICML)*, pages 1597–1607. PMLR, 2020.

[25] Divyat Mahajan, Shruti Tople, and Amit Sharma. Domain generalization using causal matching. In *International Conference on Machine Learning (ICML)*, pages 7313–7324. PMLR, 2021.

[26] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. MIT press, 2017.

[27] Bernhard Schölkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang, and Joris Mooij. On causal and anticausal learning. In *Proceedings of the 29th International Conference on Machine Learning (ICML)*, 2012.

[28] Fangrui Lv, Jian Liang, Shuang Li, Bin Zang, Chi Harold Liu, Ziteng Wang, and Di Liu. Causality inspired representation learning for domain generalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8046–8056, 2022.

[29] Maximilian Ilse, Jakub M Tomczak, Christos Louizos, and Max Welling. Diva: Domain invariant variational autoencoders. In *Medical imaging with deep learning*, pages 322–348. PMLR, 2020.

[30] Kartik Ahuja, Divyat Mahajan, Yixin Wang, and Yoshua Bengio. Interventional causal representation learning. In *International conference on machine learning*, pages 372–407. PMLR, 2023.

[31] Julius von Kügelgen, Michel Besserve, Liang Wendong, Luigi Gresele, Armin Kekić, Elias Bareinboim, David Blei, and Bernhard Schölkopf. Nonparametric identifiability of causal representations from unknown interventions. *Advances in Neural Information Processing Systems*, 36:48603–48638, 2023.

[32] Rui Gao and Anton Kleywegt. Distributionally robust stochastic optimization with wasserstein distance. *Mathematics of Operations Research*, 48(2):603–655, 2023.

[33] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):1–48, 2019.

[34] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[35] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision (ICCV)*, pages 6023–6032, 2019.

[36] Yabin Zhang, Minghan Li, Ruihuang Li, Kui Jia, and Lei Zhang. Exact feature distribution matching for arbitrary style transfer and domain generalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8035–8045, 2022.

[37] John M. Danskin. The theory of max-min, with applications. *SIAM Journal on Applied Mathematics*, 14 (4):641–664, 1966. doi: 10.1137/0114053.

[38] Gordon Christie, Neil Fendley, James Wilson, and Ryan Mukherjee. Functional map of the world. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6172–6180, 2018.

[39] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International conference on machine learning*, pages 5637–5664. PMLR, 2021.

[40] Peter Bandi, Oscar Geessink, Quirine Manson, Marcory Van Dijk, Maschenka Balkenhol, Meyke Hermsen, Babak Ehteshami Bejnordi, Byungjae Lee, Kyunghyun Paeng, Aoxiao Zhong, et al. From detection of individual metastases to classification of lymph node status at the patient level: the camelyon17 challenge. *IEEE transactions on medical imaging*, 38(2):550–560, 2018.

[41] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2020.

[42] Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 624–639, 2018.

[43] Gilles Blanchard, Aniket Anand Deshmukh, Urun Dogan, Gyemin Lee, and Clayton Scott. Domain generalization by marginal transfer learning. *Journal of machine learning research*, 22(2):1–55, 2021.

[44] Hyeonseob Nam, HyunJae Lee, Jongchan Park, Wonjun Yoon, and Donggeun Yoo. Reducing domain gap by reducing style bias. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8690–8699, 2021.

[45] Marvin Zhang, Henrik Marklund, Nikita Dhawan, Abhishek Gupta, Sergey Levine, and Chelsea Finn. Adaptive risk minimization: Learning to adapt to domain shift. *Advances in Neural Information Processing Systems*, 34:23664–23678, 2021.

[46] Zeyi Huang, Haohan Wang, Eric P. Xing, and Dong Huang. Self-challenging improves cross-domain generalization. *CoRR*, abs/2007.02454, 2020. URL `https://arxiv.org/abs/2007.02454`.

[47] Manh-Ha Bui, Toan Tran, Anh Tran, and Dinh Phung. Exploiting domain-specific features to enhance domain generalization. *Advances in Neural Information Processing Systems*, 34:21189–21201, 2021.

[48] Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee, and Sungrae Park. Swad: Domain generalization by seeking flat minima. *Advances in Neural Information Processing Systems*, 34:22405–22418, 2021.

[49] Devansh Arpit, Huan Wang, Yingbo Zhou, and Caiming Xiong. Ensemble of averages: Improving model selection and boosting performance in domain generalization. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 8265–8277. Curran Associates, Inc., 2022.

[50] Xufeng Yao, Yang Bai, Xinyun Zhang, Yuechen Zhang, Qi Sun, Ran Chen, Ruiyu Li, and Bei Yu. Pcl: Proxy-based contrastive learning for domain generalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7097–7107, 2022.

[51] Pengfei Wang, Zhaoxiang Zhang, Zhen Lei, and Lei Zhang. Sharpness-aware gradient matching for domain generalization. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3769–3778, 2023. doi: 10.1109/CVPR52729.2023.00367.

[52] Xi Yu, Huan-Hsin Tseng, Shinjae Yoo, Haibin Ling, and Yuewei Lin. Insure: an information theory inspired disentanglement and purification model for domain generalization. *IEEE Transactions on Image Processing*, 33:3508–3519, 2024.

[53] Vaibhav Rathore, Shubhranil B, Saikat Dutta, Sarthak Mehrotra, Zsolt Kira, and Biplab Banerjee. When domain generalization meets generalized category discovery: An adaptive task-arithmetic driven approach. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4905–4915, 2025.

[54] Aristotelis Ballas and Christos Diou. Gradient-guided annealing for domain generalization. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 20558–20568, 2025.

[55] Piotr Teterwak, Kuniaki Saito, Theodoros Tsiligkaridis, Kate Saenko, and Bryan Plummer. Erm++: An improved baseline for domain generalization. In *Proceedings of the Winter Conference on Applications of Computer Vision*, pages 8514–8524, 2025.

[56] Sepp Hochreiter and Jürgen Schmidhuber. Flat minima. *Neural computation*, 9(1):1–42, 1997.

[57] Miguel Monteiro, Fabio De Sousa Ribeiro, Nick Pawlowski, Daniel C Castro, and Ben Glocker. Measuring axiomatic soundness of counterfactual image models. *arXiv preprint arXiv:2303.01274*, 2023.

[58] Sander Beckers, Frederick Eberhardt, and Joseph Y Halpern. Approximate causal abstractions. In *Uncertainty in artificial intelligence*, pages 606–615. PMLR, 2020.

[59] Advait U Parulekar, Karthikeyan Shanmugam, and Sanjay Shakkottai. Pac generalization via invariant representations. In *International Conference on Machine Learning*, pages 27378–27400. PMLR, 2023.

[60] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5542–5550, 2017.

[61] Chen Fang, Ye Xu, and Daniel N Rockmore. Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. In *Proceedings of the IEEE international conference on computer vision*, pages 1657–1664, 2013.

[62] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5018–5027, 2017.

[63] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1406–1415, 2019.

[64] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*, 2020.

[65] 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*, pages 770–778, 2016.

[66] Giambattista Parascandolo, Alexander Neitz, Antonio Orvieto, Luigi Gresele, and Bernhard Schölkopf. Learning explanations that are hard to vary. *arXiv preprint arXiv:2009.00329*, 2020.

[67] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*, 2021.

[68] Juntang Zhuang, Boqing Gong, Liangzhe Yuan, Yin Cui, Hartwig Adam, Nicha C Dvornek, Sekhar Tatikonda, James s Duncan, and Ting Liu. Surrogate gap minimization improves sharpness-aware training. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=edONMAnhLu-`.

[69] Soroosh Shahtalebi, Jean-Christophe Gagnon-Audet, Touraj Laleh, Mojtaba Faramarzi, Kartik Ahuja, and Irina Rish. Sand-mask: An enhanced gradient masking strategy for the discovery of invariances in domain generalization. *arXiv preprint arXiv:2106.02266*, 2021.

[70] Guangtao Zheng, Mengdi Huai, and Aidong Zhang. Advst: Revisiting data augmentations for single domain generalization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 21832–21840, 2024.

[71] Dan Hendrycks*, Norman Mu*, Ekin Dogus Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple method to improve robustness and uncertainty under data shift. In *International Conference on Learning Representations*, 2020.

[72] Zijian Wang, Yadan Luo, Ruihong Qiu, Zi Huang, and Mahsa Baktashmotlagh. Learning to diversify for single domain generalization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 834–843, 2021.

[73] Oren Nuriel, Sagie Benaim, and Lior Wolf. Permuted adain: Reducing the bias towards global statistics in image classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9482–9491, 2021.

[74] Xiaotong Li, Yongxing Dai, Yixiao Ge, Jun Liu, Ying Shan, and LINGYU DUAN. Uncertainty modeling for out-of-distribution generalization. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=6HN7LHyzGgC.

[75] Ilke Cugu, Massimiliano Mancini, Yanbei Chen, and Zeynep Akata. Attention consistency on visual corruptions for single-source domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4165–4174, 2022.

[76] Sanqing Qu, Yingwei Pan, Guang Chen, Ting Yao, Changjun Jiang, and Tao Mei. Modality-agnostic debiasing for single domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24142–24151, 2023.

[77] Seokeon Choi, Debasmit Das, Sungha Choi, Seunghan Yang, Hyunsin Park, and Sungrack Yun. Progressive random convolutions for single domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10312–10322, 2023.

[78] Liang Chen, Yong Zhang, Yibing Song, Ying Shan, and Lingqiao Liu. Improved test-time adaptation for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24172–24182, 2023.

[79] Liang Chen, Yong Zhang, Yibing Song, Anton Van Den Hengel, and Lingqiao Liu. Domain generalization via rationale invariance. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1751–1760, 2023.

[80] Botao Ye, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Joint feature learning and relation modeling for tracking: A one-stream framework. In *European Conference on Computer Vision (ECCV)*, pages 341–357. Springer, 2022.

[81] Lianghua Huang, Xin Zhao, and Kaiqi Huang. GOT-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(5): 1562–1577, 2021.

[82] Matthias Mueller, Neil Smith, and Bernard Ghanem. A benchmark and simulator for UAV tracking. In *European Conference on Computer Vision (ECCV)*, pages 445–461. Springer, 2016.

[83] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Object tracking benchmark. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 37, pages 1834–1848, 2015.

[84] Hongwei Niu, Linhuang Xie, Jianghang Lin, and Shengchuan Zhang. Exploring semantic consistency and style diversity for domain generalized semantic segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 6245–6253, 2025.

[85] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

Appendix: Table of Contents

Theory and proofs

A Notation . . . 17
B Regularity Assumptions . . . 17
C Theoretical Analysis . . . 18
D Nuisance-Bank Proof Device for Local Monotonicity . . . 19
E A Local DRO Majorizer with a Bank-Induced Cone . . . 21
F When the Single-Normal Cone is Conservative . . . 22
G Main-Text Proofs . . . 22
H Proof of the One-Sided Soft/Gated CGS Update . . . 23
I Practical Gate Sign Convention . . . 24
J Approximately admissible intervention probes recover the cone normal . . . 24
K Positive alignment improves the true nuisance certificate . . . 26
L Causal Target/OOD Generalization Bound . . . 26
M One-step decrease of the Norm-Rescaled Practical Cone Update . . . 30
N Proof of Curvature Suppression . . . 31

Additional experiments and analyses

O Detailed Per-Domain Benchmark Results . . . 31
P Augmenting Existing DG Algorithms with CGS . . . 33
Q Comparison with Gradient-Based Methods . . . 33
R Single Source Domain Generalization Results . . . 34
S Single Object Tracking Experiments . . . 35
T Domain-Generalized Semantic Segmentation Experiments . . . 36
U Ablations & Sensitivity . . . 39

Qualitative analyses and responsible research

V Qualitative Analysis: Visualization of Intervention Probes . . . 44
W Qualitative Analysis: Loss Landscape Visualization . . . 45
X Qualitative Analysis: Visualizing Model Attention . . . 46
Y Responsible Research Considerations . . . 47

This appendix provides additional experiments, qualitative analyses, and technical proofs that complement the main paper. The sections below include the detailed benchmark breakdown, supplementary evaluations, and the proof material supporting the theoretical results.

A Notation

Spaces and variables. $s \in \mathcal{S}$: semantic component; $t \in \mathcal{T}$: nuisance component; $x = (s, t)$: input. The idealized intervention operator is $\mathcal{D}(s, t) = (s_\varnothing, t)$.

Model, loss, and gradients. $f^\theta : \mathcal{X} \to \mathbb{R}^C$ with class logit $f_c^\theta(x)$. Per-example loss $\ell(\theta; x, y)$; empirical risk $\Delta R_\nu(\theta) \coloneqq \mathbb{E}_{(x,y)\sim\nu}[\ell(\theta; x, y)]$. ERM gradient $g \coloneqq \nabla_\theta \Delta R_\nu(\theta)$. Nuisance-fit objective $J(\theta) \coloneqq \sum_i \log q_\theta(y_i \mid s_\varnothing, t_i)$ and nuisance-certificate gradient / cone normal $r \coloneqq \nabla_\theta J(\theta)$.

True and estimated cone normals. If $r \neq 0$, define the unit *true* cone normal by $u \coloneqq r/\|r\|_2$. For a practical estimated normal $\tilde{r}$, write $\hat{u} \coloneqq \tilde{r}/\|\tilde{r}\|_2$. The scalar harmful-alignment coordinate is $a \coloneqq \langle g, \hat{u}\rangle$.

Nuisance geometry. Metric on nuisance space: $d_T(t, t')$. Transport cost between inputs:

$$c\big((s, t), (s', t')\big) = \begin{cases} d_T(t, t'), & s = s', \\ +\infty, & s \neq s'. \end{cases}$$

Wasserstein ball radius ρ defines the DRO set $\{Q : W_c(Q, P) \leq \rho\}$.

Robust risks. Worst-case risk (primal) $\mathcal{R}_\rho(\theta) = \sup_{Q:W_c(Q,P)\leq\rho} \mathbb{E}_Q[\ell(\theta; X, Y)]$; dual form uses a penalty $\lambda \geq 0$ as in Eq. (7). We write $F(\theta, \lambda) \coloneqq \lambda\rho + \frac{1}{n}\sum_i h_i(\theta, \lambda)$ with $h_i(\theta, \lambda) := \sup_{t'\in\mathcal{T}}(\ell(\theta; s_i, t', y_i) - \lambda d_T(t', t_i))$, and denote the active adversary set by $\mathcal{A}_i(\theta, \lambda) := \arg\max_{t'\in\mathcal{T}}(\ell(\theta; s_i, t', y_i) - \lambda d_T(t', t_i))$. The bank-support nuisance surcharge is $\Gamma_\theta(\Delta\theta) \coloneqq \frac{1}{n}\sum_i \sup_{\tau\in\mathcal{B}_{y_i}} \langle \nabla_\theta \ell(\theta; s_i, \tau, y_i) - \nabla_\theta \ell(\theta; s_i, t_i, y_i), \Delta\theta\rangle$.

Operators and conventions. $\langle u, v\rangle$: inner product; $\|v\|$: Euclidean norm; $\mathbb{I}\{\cdot\}$: indicator. $[z]_+ \coloneqq \max\{z, 0\}$; $\arg\min$ and $\arg\max$ denote optimizer sets.

Constants and hyperparameters. L: loss-level Lipschitz constant in nuisance space; M: nuisance-gradient Lipschitz constant in nuisance space; β: parameter-space smoothness; ε_{cov}: bank-coverage radius; η: learning rate; $\tau_{\cos} \in [0, 1]$: theoretical cosine-margin threshold; $\tau_{\text{dp}} \geq 0$: practical raw dot-score threshold used in the released classification code; $\gamma \in [0, 1)$: proximal projection strength in the theory; practical sweeps may also report $\gamma > 1$ as an over-relaxation multiplier, while the default reported configuration uses $\gamma = 0.5$.

B Regularity Assumptions

Assumption B.1 (Regularity Conditions)**.** We assume the following for all inputs $x = (s, t)$, parameters θ, and labels y.

A1 *(Loss-level Lipschitzness in the nuisance component).* There exists $L > 0$ such that for any $t, t' \in T$,

$$\big|\ell\big(f^\theta(s, t), y\big) - \ell\big(f^\theta(s, t'), y\big)\big| \leq L\, d_T(t, t').$$

A2 *(Local θ-smoothness on the parameter region visited by training).* There exists a convex set $\Theta \subseteq \mathbb{R}^d$ that contains all parameter values considered in the analysis. For every fixed input representation ξ and label y, the loss map $\theta \mapsto \ell(f^\theta(\xi), y)$ is continuously differentiable on an open neighborhood of Θ and has β-Lipschitz gradient on Θ:

$$\|\nabla_\theta \ell(f^{\theta'}(\xi), y) - \nabla_\theta \ell(f^\theta(\xi), y)\| \leq \beta\|\theta' - \theta\|, \qquad \forall \theta, \theta' \in \Theta.$$

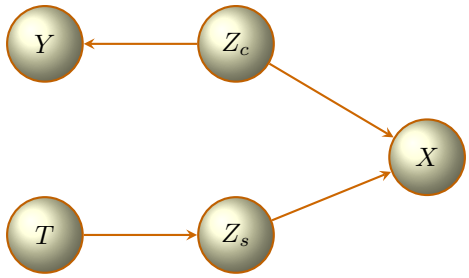


Figure 3: Structural causal model used by the theory: the label Y is generated from causal features Z_c, the nuisance source T influences spurious features Z_s, and the observed input X depends on both.

For the logit-gain lemma, the scalar logit maps $\theta \mapsto f_c^\theta(\xi)$ satisfy the same local smoothness condition, possibly with a smaller problem-dependent constant; we use β for the maximum of the two constants.

A3 *(Attainment and local Danskin upper modeling).* For each sample, the inner supremum over t' in the dual robust objective is attained on the local nuisance set considered by the adversary. Let $\mathcal{A}_i(\theta, \lambda)$ be the base-point active set of maximizers for $h_i(\theta, \lambda)$. For every base point (θ, λ) on the training path there exists a modulus $\omega_{i,\theta,\lambda}(r) \to 0$ as $r \downarrow 0$ such that, for all sufficiently small $\Delta\theta$,

$$h_i(\theta+\Delta\theta, \lambda) \le h_i(\theta, \lambda)+ \sup_{t^* \in \mathcal{A}_i(\theta,\lambda)} \langle \nabla_\theta \ell(\theta; s_i, t^*, y_i), \Delta\theta\rangle+\frac{\beta}{2}\|\Delta\theta\|^2+\omega_{i,\theta,\lambda}(\|\Delta\theta\|)\|\Delta\theta\|.$$

This assumption is the finite-step version of the local Danskin expansion. It is automatic with $\omega = 0$ under locally stable finite active sets; otherwise it records the small active-set drift that a pointwise maximum can introduce. Gradients are not propagated through the maximizing nuisance argument itself.

A4 *(Nuisance-gradient Lipschitzness).* There exists $M > 0$ such that for every training sample i and all $t', \tilde{t} \in \mathcal{T}$,

$$\|\nabla_\theta \ell(\theta; s_i, t', y_i) - \nabla_\theta \ell(\theta; s_i, \tilde{t}, y_i)\| \le M\, d_T(t', \tilde{t}).$$

A5 *(Active-adversary bank coverage).* For the base point θ and the multiplier selected by the local argument (an unrestricted minimizer $\bar{\lambda} \in \arg\min_{\lambda \ge 0} F(\theta, \lambda)$, or the restricted minimizer $\bar{\lambda} \in \arg\min_{0 \le \lambda \le \Lambda} F(\theta, \lambda)$ in Appendix L), every active adversarial nuisance $t^* \in \mathcal{A}_i(\theta, \bar{\lambda})$ is covered by the class-conditional bank: there exists a bank pair $(\tau, t_\tau) \in \mathcal{B}_{y_i}$ such that $d_T(t^*, t_\tau) \le \varepsilon_{\text{cov}}$.

C Theoretical Analysis

Having derived the CGS update rule and its practical implementation, we now provide a brief theoretical analysis of its properties. We demonstrate that CGS is a principled descent method for the empirical risk under the cone-relaxed update and offer a geometric interpretation of its effect on the loss landscape. Detailed proofs for the following results are provided in the Appendix.

C.1 One-Step Descent Guarantee

Our primary theoretical result confirms that the raw norm-rescaled cone update used in the practical rule makes guaranteed progress on the empirical risk when that objective is smooth. This establishes CGS as a valid optimization procedure for the local cone-relaxed update geometry derived in the main text.

Theorem C.1 (One-Step Decrease for the Norm-Rescaled Practical Cone Update)**.** Assume ΔR_ν is L_θ-smooth as in (40). Let $g := \nabla_\theta \Delta R_\nu(\theta)$, let $\hat{u}$ be an estimated unit cone normal, define $a := \langle g, \hat{u}\rangle$,

$I_- \coloneqq \mathbb{I}\{a < 0\}$, and

$$v \coloneqq g - \gamma I_- \, a \, \hat{u}, \qquad \Delta\theta^\star \;=\; -\eta \frac{\|g\|_2}{\|v\|_2} v, \qquad q \coloneqq \frac{I_- a^2}{\|g\|_2^2} \in [0,1].$$

Then

$$\Delta R_\nu(\theta + \Delta\theta^\star) \;\le\; \Delta R_\nu(\theta) \;-\; \eta \, \|g\|_2^2 \, \frac{1 - \gamma q}{\sqrt{1 - \gamma(2-\gamma) q}} \;+\; \frac{L_\theta \, \eta^2}{2} \|g\|_2^2. \tag{15}$$

Define $m_\gamma \coloneqq \frac{2\sqrt{1-\gamma}}{2-\gamma}$. If $\eta \le m_\gamma / L_\theta$, then $\Delta R_\nu(\theta + \Delta\theta^\star) \le \Delta R_\nu(\theta) - \frac{\eta m_\gamma}{2} \|g\|_2^2$.

Proof. See Appendix M.

The proof follows by applying the standard descent lemma to the norm-rescaled practical update. The linear term depends on the cosine-adjusted progress factor $\frac{1-\gamma q}{\sqrt{1-\gamma(2-\gamma)q}}$, while the quadratic term reflects the restored ERM norm budget $\|\Delta\theta^\star\|_2 = \eta \|g\|_2$.

C.2 Geometric Insight: Curvature Suppression

Beyond guaranteeing descent, the hard-projection geometry has a useful curvature interpretation. It is widely observed that flatter minima in the loss landscape often correspond to more generalizable solutions [56]. We can show that the active hard-projection step removes the cone-normal direction and therefore restricts optimization to a tangent subspace with lower maximal curvature whenever the cone normal overlaps the sharpest eigendirection.

Proposition C.2 (Curvature Suppression)**.** Let $H(\theta)$ be the Hessian of the empirical risk, with largest eigenvalue $\lambda_1(\theta)$ and corresponding eigenvector $v_1(\theta)$. When the hard CGS projection is active along a unit cone normal u, the maximum curvature on the cone-boundary tangent subspace $u^\perp$ is bounded by

$$\max_{v \perp u, \|v\|=1} v^\top H(\theta) v \le \lambda_1(\theta) - (\lambda_1(\theta) - \lambda_2(\theta)) \langle u, v_1(\theta) \rangle^2, \tag{16}$$

where $\lambda_2(\theta)$ is the second-largest eigenvalue of $H(\theta)$.

This result shows that if the cone normal has any alignment with the sharpest direction of the loss landscape, then the active hard projection strictly reduces the worst-case curvature available on the tangent subspace. It therefore supplies a geometric explanation for why cone-feasible directions can be flatter than the unconstrained top-eigenvector direction.

D Nuisance-Bank Proof Device for Local Monotonicity

The paper states the local Danskin majorizer and then compresses it to a tractable single-normal cone. This section supplies the nuisance-bank bookkeeping device that supports the bank-induced nuisance surcharge appearing in that majorizer.

Definition D.1 (Nuisance Banks)**.** For each class c, define a *nuisance bank* as a collection of idealized intervention-transformed inputs together with their associated latent nuisance anchors:

$$\mathcal{B}_c \;\coloneqq\; \{\, (\tau_j, t_j) \;:\; \tau_j = \mathcal{D}(x_j),\; x_j = (s_j, t_j),\; y_j = c \,\}.$$

We write $(\tau, t_\tau) \in \mathcal{B}_c$ to denote a class-c bank pair, where τ is the observable intervention-transformed exemplar and t_τ is its latent nuisance anchor.

Lemma D.2 (Derivative Gains at Nuisance Exemplars)**.** Consider an update to the model parameters, $\theta^+ = \theta + \alpha u$, for some direction u and step size $\alpha > 0$, and assume $\theta, \theta^+ \in \Theta$ as in Assumption B.1(**A2**). For any intervention-transformed exemplar τ appearing in a bank pair $(\tau, t_\tau) \in \mathcal{B}_c$, the change in the nuisance-only logit for class c, denoted $\Delta_c(\tau) \coloneqq f_c^{\theta^+}(\tau) - f_c^\theta(\tau)$, is lower-bounded by

$$\Delta_c(\tau) \ge \alpha \langle \nabla_\theta f_c^\theta(\tau), u \rangle - \frac{\beta}{2} \alpha^2 \|u\|^2, \tag{17}$$

where $f_c^\theta(\tau)$ denotes the class-c logit evaluated on the intervention-transformed input τ, and β is the smoothness constant from Assumption B.1.

Proof. Fix a bank pair $(\tau, t_\tau) \in \mathcal{B}_c$ and define $g_\tau(\theta) \coloneqq f_\theta^c(\tau)$. Let $\Delta\theta \coloneqq \theta^+ - \theta = \alpha u$. By Assumption B.1(**A2**), g_τ is continuously differentiable on an open neighborhood of a convex set Θ containing θ and θ^+, and ∇g_τ is β-Lipschitz on Θ. Since Θ is convex, the line segment $\theta_t \coloneqq \theta + t\Delta\theta$ for $t \in [0,1]$ lies in Θ. By the fundamental theorem of calculus,

$$g_\tau(\theta + \Delta\theta) - g_\tau(\theta) = \int_0^1 \big\langle \nabla g_\tau(\theta_t),\, \Delta\theta \big\rangle \, dt.$$

Adding and subtracting $\nabla g_\tau(\theta)$ inside the integrand yields

$$g_\tau(\theta + \Delta\theta) - g_\tau(\theta) = \big\langle \nabla g_\tau(\theta),\, \Delta\theta \big\rangle + \int_0^1 \big\langle \nabla g_\tau(\theta_t) - \nabla g_\tau(\theta),\, \Delta\theta \big\rangle \, dt.$$

For each $t \in [0,1]$, Cauchy–Schwarz and the β-Lipschitzness of ∇g_τ on Θ imply

$$\big\langle \nabla g_\tau(\theta_t) - \nabla g_\tau(\theta),\, \Delta\theta \big\rangle \geq -\beta t \, \|\Delta\theta\|^2.$$

Integrating over $t \in [0,1]$ gives

$$\int_0^1 \big\langle \nabla g_\tau(\theta_t) - \nabla g_\tau(\theta),\, \Delta\theta \big\rangle \, dt \geq -\frac{\beta}{2} \, \|\Delta\theta\|^2.$$

Combining the above displays and substituting $\Delta\theta = \alpha u$ yields (17). □

Lemma D.3 (Two-sided loss deviation in the target component)**.** Assume A1. Then for any $t, t' \in T$,

$$\big|\ell\big(f^\theta(s,t), y\big) - \ell\big(f^\theta(s,t'), y\big)\big| \;\leq\; L\, d_T(t,t')\,. \tag{18}$$

Equivalently, for any $\rho > 0$,

$$\sup_{d_T(t,t') \leq \rho} \big|\ell\big(f^\theta(s,t), y\big) - \ell\big(f^\theta(s,t'), y\big)\big| \;\leq\; L\,\rho\,.$$

Proof. The two-sided inequality is exactly Assumption B.1(**A1**). The radius form follows by taking the supremum over $\{t' : d_T(t,t') \leq \rho\}$. □

Vulnerable regions in nuisance space. Fix a class c and radius $\rho > 0$. Let the class-c nuisance bank be $\mathcal{B}_c$ (Definition D.1). For a parameter update $\theta^+ = \theta + \alpha u$, recall $\Delta_c(\tau) \coloneqq f_c^{\theta^+}(\tau) - f_c^\theta(\tau)$. Define the *vulnerable region* for class c by

$$U_c(\theta;\rho) \;\coloneqq \bigcup_{(\tau,t_\tau)\in\mathcal{B}_c:\ \Delta_c(\tau)>0} \{\, t' \in \mathcal{T} :\ d_T(t_\tau, t') \leq \rho \,\}.$$

By Lemma D.3, any positive logit gain at a bank exemplar propagates to the entire d_T-ball of radius ρ around its associated nuisance anchor. Enlarging $U_c(\theta;\rho)$ therefore enlarges the set on which the inner adversary in Eq. (7) can raise the loss at low transport cost.

Proposition D.4 (Dual-consistent robust loss bound at radius ρ)**.** Fix $x = (s,t)$ and $\rho > 0$. Define the radius-restricted envelope

$$\mathcal{L}_{\text{ball}}(\theta; x, \rho) := \sup_{d_T(t,t') \leq \rho} \ell(f^\theta(s,t'), y).$$

Under A1,

$$\mathcal{L}_{\text{ball}}(\theta; x, \rho) \leq \ell(f^\theta(s,t), y) + L\rho. \tag{19}$$

Moreover, the Wasserstein dual inner term uses the negative transport penalty

$$\sup_{t' \in \mathcal{T}} \{\ell(f^\theta(s,t'), y) - \lambda d_T(t',t)\}.$$

If $\lambda \geq L$ and the nuisance space is unbounded only through d_T, this penalized inner term is upper-bounded by the clean loss $\ell(f^\theta(s,t), y)$. Thus the radius envelope and the dual penalty are sign-consistent: the radius view exposes where nearby nuisance perturbations can be profitable, while the dual penalty controls how far the adversary can move.

Proof. For any t' with $d_T(t,t') \leq \rho$, Lemma D.3 implies

$$\ell(f^\theta(s,t'), y) \leq \ell(f^\theta(s,t), y) + L\rho.$$

Taking the supremum over the ρ-ball proves (19). For the penalized dual term, A1 gives $\ell(f^\theta(s,t'), y) - \lambda d_T(t',t) \;\leq\; \ell(f^\theta(s,t), y) + (L - \lambda) d_T(t',t)$, which is at most the clean loss when $\lambda \geq L$. □

E A Local DRO Majorizer with a Bank-Induced Cone

Theorem E.1 (Full Local Danskin DRO Majorizer with Bank-Induced Nuisance Surcharge)**.** Let

$$F(\theta,\lambda) := \lambda\rho + \frac{1}{n}\sum_{i=1}^{n} h_i(\theta,\lambda), \qquad h_i(\theta,\lambda) := \sup_{t'\in\mathcal{T}} \big(\ell(\theta; s_i, t', y_i) - \lambda d_T(t', t_i)\big),$$

so that $\mathcal{R}_\rho(\theta) = \min_{\lambda\ge 0} F(\theta,\lambda)$. Fix $\bar\lambda \in \arg\min_{\lambda\ge 0} F(\theta,\lambda)$ and define

$$\mathcal{A}_i(\theta,\bar\lambda) := \arg\max_{t'\in\mathcal{T}} \big(\ell(\theta; s_i, t', y_i) - \bar\lambda d_T(t', t_i)\big).$$

For class-wise nuisance banks $\{\mathcal{B}_c\}_c$, define

$$\Gamma_\theta(\Delta\theta) := \frac{1}{n}\sum_{i=1}^{n} \sup_{(\tau,t_\tau)\in\mathcal{B}_{y_i}} \langle \nabla_\theta \ell(\theta; s_i, t_\tau, y_i) - \nabla_\theta \ell(\theta; s_i, t_i, y_i), \Delta\theta\rangle .$$

Let $g = n^{-1}\sum_i \nabla_\theta \ell(\theta; s_i, t_i, y_i)$. Under Assumption B.1, every sufficiently small perturbation $\Delta\theta$ satisfies

$$\mathcal{R}_\rho(\theta+\Delta\theta) \le \mathcal{R}_\rho(\theta) + \langle g, \Delta\theta\rangle + [\Gamma_\theta(\Delta\theta)]_+ + M\varepsilon_{\mathrm{cov}}\|\Delta\theta\| + \frac{\beta}{2}\|\Delta\theta\|^2 + \omega_\theta(\|\Delta\theta\|)\|\Delta\theta\|.$$

Consequently, the first-order robust-feasible cone induced by nuisance-only DRO is

$$\mathcal{C}_\theta^{\mathrm{bank}} := \{\Delta\theta \in \mathbb{R}^d : \Gamma_\theta(\Delta\theta) \le 0\}.$$

Proof of Theorem E.1. This section proves the local DRO bridge used in the main text. The key point is that nuisance-only DRO yields a local Danskin upper model whose nuisance term is a support function over active nuisance gradient increments, plus an explicit vanishing modulus for active-set drift. The single-normal cone used by CGS is then presented as a tractable compression of this richer bank-induced geometry.

Fix a base point θ and choose $\bar\lambda \in \arg\min_{\lambda\ge 0} F(\theta,\lambda)$, so $\mathcal{R}_\rho(\theta) = F(\theta,\bar\lambda)$. By optimality of the outer minimization at the perturbed point,

$$\mathcal{R}_\rho(\theta+\Delta\theta) \le F(\theta+\Delta\theta, \bar\lambda).$$

It therefore suffices to upper-bound $F(\theta+\Delta\theta,\bar\lambda) - F(\theta,\bar\lambda)$.

For each sample i, write

$$\phi_i(\vartheta, t') := \ell(\vartheta; s_i, t', y_i) - \bar\lambda d_T(t', t_i), \qquad h_i(\vartheta,\bar\lambda) = \sup_{t'\in\mathcal{T}} \phi_i(\vartheta, t').$$

By Assumption B.1(**A3**), the local directional upper model of h_i is governed by the active set $\mathcal{A}_i(\theta,\bar\lambda)$ at the base point, with a vanishing active-set-drift modulus. Hence for sufficiently small $\Delta\theta$,

$$h_i(\theta+\Delta\theta,\bar\lambda) \le h_i(\theta,\bar\lambda) + \sup_{t^*\in\mathcal{A}_i(\theta,\bar\lambda)} \langle \nabla_\theta \ell(\theta; s_i, t^*, y_i), \Delta\theta\rangle + \frac{\beta}{2}\|\Delta\theta\|^2 + \omega_{i,\theta,\bar\lambda}(\|\Delta\theta\|)\|\Delta\theta\|.$$

By Assumption B.1(**A5**), each active adversary t^* is covered by some bank pair $(\tau, t_\tau) \in \mathcal{B}_{y_i}$ with $d_T(t^*, t_\tau) \le \varepsilon_{\mathrm{cov}}$. Assumption B.1(**A4**) therefore implies

$$\big\|\nabla_\theta \ell(\theta; s_i, t^*, y_i) - \nabla_\theta \ell(\theta; s_i, t_\tau, y_i)\big\| \le M\,\varepsilon_{\mathrm{cov}},$$

and hence

$$\langle \nabla_\theta \ell(\theta; s_i, t^*, y_i), \Delta\theta\rangle \le \langle \nabla_\theta \ell(\theta; s_i, t_\tau, y_i), \Delta\theta\rangle + M\,\varepsilon_{\mathrm{cov}}\,\|\Delta\theta\|.$$

Adding and subtracting the clean gradient $\nabla_\theta \ell(\theta; s_i, t_i, y_i)$ and then taking the supremum over $(\tau, t_\tau) \in \mathcal{B}_{y_i}$ yields

$$\begin{aligned} h_i(\theta+\Delta\theta,\bar\lambda) \le\; & h_i(\theta,\bar\lambda) + \langle \nabla_\theta \ell(\theta; s_i, t_i, y_i), \Delta\theta\rangle \\ & + \sup_{(\tau,t_\tau)\in\mathcal{B}_{y_i}} \Big\langle \nabla_\theta \ell(\theta; s_i, t_\tau, y_i) - \nabla_\theta \ell(\theta; s_i, t_i, y_i), \Delta\theta \Big\rangle \\ & + M\,\varepsilon_{\mathrm{cov}}\,\|\Delta\theta\| + \frac{\beta}{2}\|\Delta\theta\|^2 + \omega_{i,\theta,\bar\lambda}(\|\Delta\theta\|)\|\Delta\theta\|. \end{aligned}$$

Averaging over i gives

$$F(\theta+\Delta\theta,\bar\lambda) \le F(\theta,\bar\lambda) + \langle g, \Delta\theta\rangle + \Gamma_\theta(\Delta\theta) + M\,\varepsilon_{\mathrm{cov}}\,\|\Delta\theta\| + \frac{\beta}{2}\|\Delta\theta\|^2 + \omega_\theta(\|\Delta\theta\|)\|\Delta\theta\|.$$

Since $\Gamma_\theta(\Delta\theta) \le [\Gamma_\theta(\Delta\theta)]_+$, Theorem 2.1 follows immediately, with $\omega_\theta(r) = n^{-1}\sum_i \omega_{i,\theta,\bar\lambda}(r)$ and $\omega_\theta(r) \to 0$ as $r \downarrow 0$. □

Corollary (bank-induced first-order cone). The set

$$\mathcal{C}_\theta^{\mathrm{bank}} := \{\Delta\theta : \Gamma_\theta(\Delta\theta) \le 0\}$$

is the first-order robust-feasible cone associated with the local majorizer above. Inside this cone, the nuisance surcharge is inactive to first order up to the explicit coverage, vanishing Danskin, and quadratic remainders.

Remark (why CGS uses a single normal). The bank-induced cone is generally defined by a support function rather than one half-space, so projecting onto it exactly would require the whole active bank. The single-normal cone $\mathcal{C}_\theta = \{\Delta\theta : \langle r, \Delta\theta\rangle \le 0\}$ used by CGS is therefore an algorithmic compression of the exact geometry. It becomes an exact reduction only under additional domination assumptions. This is the precise boundary between the exact DRO statement and the tractable algorithmic approximation used in the main text.

F When the Single-Normal Cone is Conservative

The main algorithm uses a single normal because it is the only object that can be estimated cheaply from the destroyed branch. The following proposition makes explicit the condition under which this compression is conservative for the richer bank cone. Let the bank surcharge be written as a support function

$$\Gamma_\theta(\Delta\theta) = \sup_{b \in \mathcal{G}_\theta} \langle b, \Delta\theta\rangle,$$

where $\mathcal{G}_\theta$ denotes the active nuisance-gradient increments collected by the bank construction.

Proposition F.1 (Dominating-normal sufficient condition)**.** Assume there are constants $\kappa > 0$ and $\xi \ge 0$ such that, for every active bank normal $b \in \mathcal{G}_\theta$ and every update $\Delta\theta$ in the local trust region considered by Theorem 2.1,

$$\langle b, \Delta\theta\rangle \le \kappa \langle r, \Delta\theta\rangle + \xi \|\Delta\theta\|. \tag{20}$$

Then the margin-shrunken single-normal cone

$$\mathcal{C}_\theta(\xi, \kappa) = \left\{\Delta\theta : \langle r, \Delta\theta\rangle \le -\frac{\xi}{\kappa}\|\Delta\theta\|\right\}$$

is contained in the bank cone $\mathcal{C}_\theta^{\mathrm{bank}}$. In the exact domination case $\xi = 0$, the ordinary single-normal cone $\{\Delta\theta : \langle r, \Delta\theta\rangle \le 0\}$ is conservative for the bank cone.

Proof. For any $\Delta\theta \in \mathcal{C}_\theta(\xi, \kappa)$ and any $b \in \mathcal{G}_\theta$, Eq. (20) gives

$$\langle b, \Delta\theta\rangle \le \kappa\left(-\frac{\xi}{\kappa}\|\Delta\theta\|\right) + \xi\|\Delta\theta\| = 0.$$

Taking the supremum over $b \in \mathcal{G}_\theta$ yields $\Gamma_\theta(\Delta\theta) \le 0$, hence $\Delta\theta \in \mathcal{C}_\theta^{\mathrm{bank}}$. The case $\xi = 0$ is immediate. □

This proposition is not an extra assumption used by CGS. It is a calibration statement: when active nuisance-bank normals are dominated by the destroyed-view certificate, the single-normal projection is a conservative surrogate for the exact bank geometry; otherwise it remains a computationally cheap compression whose quality is governed by the admissibility and alignment results in Appendices J–K.

G Moved Main-Text Proofs

Proof of Proposition 2.3. The claim is the first-order expansion of the differentiable scalar map J at θ:

$$J(\theta + \Delta\theta) = J(\theta) + \langle r, \Delta\theta\rangle + o(\|\Delta\theta\|).$$

Substituting the definition of $\mathcal{C}_\theta$ gives the stated sign conclusions. □

Proof of Theorem 2.4. The objective in Eq. (9) is strictly convex and the constraint is linear, so the optimizer is unique. The KKT conditions are

$$\Delta\theta^\star + \eta g + \alpha r = 0, \qquad \alpha \geq 0, \qquad \langle r, \Delta\theta^\star \rangle \leq 0, \qquad \alpha \langle r, \Delta\theta^\star \rangle = 0.$$

If the ERM step is already feasible, $\langle r, -\eta g \rangle \leq 0$ or equivalently $\langle g, r \rangle \geq 0$, then $\alpha = 0$ and $\Delta\theta^\star = -\eta g$.

If instead $\langle g, r \rangle < 0$, the constraint is active and $\langle r, \Delta\theta^\star \rangle = 0$. Solving

$$\Delta\theta^\star = -\eta g - \alpha r, \qquad 0 = \langle r, \Delta\theta^\star \rangle = -\eta \langle r, g \rangle - \alpha \|r\|_2^2$$

gives

$$\alpha = -\eta \frac{\langle r, g \rangle}{\|r\|_2^2},$$

and therefore

$$\Delta\theta^\star = -\eta \left(g - \frac{\langle g, r \rangle}{\|r\|_2^2} r \right) = -\eta (g - \langle g, \hat{r} \rangle \hat{r}),$$

where $\hat{r} = r/\|r\|_2$. This is the stated piecewise solution. □

Proof of Proposition 2.8. The objective in Eq. (14) is a sum of independent layerwise terms, so the optimizer is obtained by solving each block separately. Each block is exactly the one-sided cone-relaxation problem from Proposition 2.5, with $(g, \hat{u}, \eta, \mu)$ replaced by $(g_l, \hat{u}_l, \eta_l, \mu_l)$, which yields

$$\Delta\theta_l^\star = -\eta_l \big(g_l - \gamma_l I_-^{(l)} a_l \hat{u}_l \big).$$

The cosine-margin relation follows immediately from the definition of

$$\rho_l = \frac{\langle g_l, \tilde{r}_l \rangle}{\|g_l\|_2 \|\tilde{r}_l\|_2 + \epsilon}$$

and the normalization $\hat{u}_l = \tilde{r}_l / \|\tilde{r}_l\|_2$. □

H Proof of the One-Sided Soft/Gated CGS Update (cf. Algorithm 1)

We show that a one-sided proximal penalty against cone violation yields the closed-form soft/gated CGS step used in practice. Let $g \coloneqq \nabla_\theta \Delta R_\nu(\theta)$ be the ERM gradient, let $\tilde{r} \in \mathbb{R}^d$ denote the estimated cone normal, and let $\hat{u} \coloneqq \tilde{r}/\|\tilde{r}\|_2$ be its unit normalization. For step size $\eta > 0$ and penalty $\mu \geq 0$, consider

$$\min_{\Delta\theta \in \mathbb{R}^d} \quad \frac{1}{2\eta} \|\Delta\theta + \eta\, g\|_2^2 \;+\; \frac{\mu}{2} [\langle \Delta\theta, \hat{u} \rangle]_+^2. \tag{21}$$

Write $a \coloneqq \langle g, \hat{u} \rangle$ and decompose any candidate update as $\Delta\theta = z + x\hat{u}$ with $z \perp \hat{u}$. Since the penalty depends only on the scalar component $x = \langle \Delta\theta, \hat{u} \rangle$, the orthogonal part is minimized by

$$z^\star = -\eta \big(g - a\hat{u} \big).$$

Substituting this into Eq. (21) reduces the problem to the scalar optimization

$$\min_{x \in \mathbb{R}} \quad \frac{1}{2\eta} (x + \eta a)^2 + \frac{\mu}{2} [x]_+^2.$$

If $a \geq 0$, then the unconstrained minimizer of the quadratic term is $x^\star = -\eta a \leq 0$, so the penalty is inactive and $\Delta\theta^\star = -\eta g$.

If $a < 0$, then the minimizer lies on the branch $x > 0$, where the objective is differentiable and

$$\frac{1}{\eta} (x + \eta a) + \mu x = 0.$$

Solving gives

$$x^\star = -\frac{\eta a}{1 + \eta\mu} > 0.$$

Hence

$$\begin{aligned}\Delta\theta^\star &= -\eta\big(g - a\hat{u}\big) - \frac{\eta a}{1+\eta\mu}\hat{u} \\ &= -\eta\left(g - \gamma a\hat{u}\right),\end{aligned} \tag{22}$$

where

$$\gamma \coloneqq \frac{\eta\mu}{1+\eta\mu} \in [0,1).$$

Combining the two cases gives

$$\Delta\theta^\star = -\eta\Big(g - \gamma\,\mathbb{I}\{\langle g,\hat{u}\rangle < 0\}\,\langle g,\hat{u}\rangle\,\hat{u}\Big),$$

which is the soft/gated CGS step. Equivalently,

$$\Delta\theta^\star = -\eta\Big(g - \gamma\,\mathbb{I}\{\langle g,\tilde{r}\rangle < 0\}\,\frac{\langle g,\tilde{r}\rangle}{\|\tilde{r}\|_2^2}\,\tilde{r}\Big).$$

This matches the practical relaxation underlying Algorithm 1.

I Practical Gate Sign Convention

Algorithm 1 is written in optimizer-gradient form, whereas the theory defines $r = \nabla_\theta J(\theta)$ as the ascent direction of nuisance-based label fit. The practical probe gradient $\tilde{h}_l = \nabla_{\theta_l}\ell(f_\theta(\mathcal{P}(x)), y)$ is the destroyed-view cross-entropy gradient, so it estimates the opposite sign of the nuisance-fit log-likelihood ascent normal. Thus the theoretical harmful condition $\langle g_l, r_l\rangle < 0$ is implemented as $\langle g_l, \tilde{h}_l\rangle > 0$: when the clean gradient aligns with the destroyed-view loss gradient, the clean descent step $-g_l$ would increase the nuisance-fit certificate to first order. Equivalently, the practical notation replaces $\hat{u}_l$ by $-\tilde{h}_l/\|\tilde{h}_l\|_2$, so the harmful branch in Algorithm 1 appears as $\langle g_l, \tilde{h}_l\rangle > \tau_{\mathrm{dp}}$ and the update subtracts the projection onto $\tilde{h}_l$. The scalar ρ_l is the corresponding cosine alignment and is useful for diagnostics or a normalized-gate variant; the reported experiments use the raw dot-score threshold τ_{dp} as a numerical sparsity margin. The method is minibatch stochastic: g_l and $\tilde{h}_l$ are gradients of averaged clean and destroyed minibatch losses.

J Approximately admissible intervention probes recover the cone normal

Exact latent counterfactual recovery is too strong a target for high-dimensional image probes, and exact causal-representation identification typically requires richer interventional structure than the single-probe regime considered here [57, 30, 31]. Following the spirit of approximate causal abstractions [58], we instead ask for an *approximately admissible intervention probe*: a probe-transformed view should stay close to the ideal nuisance-only view in the representation coordinates that matter for the optimizer. This is already enough to recover the cone normal approximately and, through the existing positive-alignment bridge, to justify the practical correction. Approximate rather than exact invariance can still be mathematically meaningful [59].

Fix a model state θ, a labeled example (x, y), and an ideal nuisance-only view x^ν from the theory-level intervention model. At the representation level used by the probe argument, write

$$h_\theta(x) = \big(h_{s,\theta}(x), h_{t,\theta}(x)\big) \in \mathcal{Z}_s \times \mathcal{Z}_t,$$

and let $\mathcal{P}(x)$ denote the practical probe-transformed input. We model the loss gradient in these coordinates by a map $G_\theta : \mathcal{Z}_s \times \mathcal{Z}_t \times \mathcal{Y} \to \mathbb{R}^d$ satisfying

$$\nabla_\theta\ell(\theta; x, y) = G_\theta\big(h_{s,\theta}(x), h_{t,\theta}(x); y\big).$$

Define the ideal and practical cone normals in the same coordinates:

$$r \coloneqq -G_\theta\big(h_{s,\theta}(x^\nu), h_{t,\theta}(x^\nu); y\big), \qquad \tilde{r} \coloneqq -G_\theta\big(h_{s,\theta}(\mathcal{P}(x)), h_{t,\theta}(\mathcal{P}(x)); y\big).$$

Whenever $r \neq 0$, let $u \coloneqq r/\|r\|_2$ and, when $\tilde{r} \neq 0$, let $\hat{u} \coloneqq \tilde{r}/\|\tilde{r}\|_2$.

Assumption J.1 (Local representation-space regularity for the probe bridge)**.** At the current model state and representation level under study, there exist constants $L_s, L_t \geq 0$ and $\underline{r} > 0$ such that

$$\|G_\theta(z_s, z_t; y) - G_\theta(z'_s, z'_t; y)\|_2 \leq L_s\|z_s - z'_s\|_2 + L_t\|z_t - z'_t\|_2,$$
$$\|r\|_2 \geq \underline{r}.$$

Definition J.2 (Approximately admissible intervention probe)**.** The intervention probe $\mathcal{P}$ is $(\varepsilon_s, \varepsilon_t)$-admissible for (x, y) relative to the ideal nuisance-only view x^ν if

$$\|h_{s,\theta}(\mathcal{P}(x)) - h_{s,\theta}(x^\nu)\|_2 \leq \varepsilon_s, \qquad \|h_{t,\theta}(\mathcal{P}(x)) - h_{t,\theta}(x^\nu)\|_2 \leq \varepsilon_t.$$

The first term measures semantic leakage and the second measures nuisance distortion.

Proposition J.3 (Admissibility implies cone-normal accuracy)**.** Under Assumption J.1 and Definition J.2,

$$\|\tilde{r} - r\|_2 \leq L_s\varepsilon_s + L_t\varepsilon_t.$$

In particular, exact pixel-space counterfactual generation is unnecessary: bounded semantic leakage and nuisance distortion already control the cone-normal error in the optimizer coordinates.

Proof. Apply Assumption J.1 with $(z_s, z_t) = \big(h_{s,\theta}(\mathcal{P}(x)), h_{t,\theta}(\mathcal{P}(x))\big)$ and $(z'_s, z'_t) = \big(h_{s,\theta}(x^\nu), h_{t,\theta}(x^\nu)\big)$, then use Definition J.2. □

Corollary J.4 (Normalized alignment from admissibility)**.** Let $\delta_r \coloneqq L_s\varepsilon_s + L_t\varepsilon_t$. If $\delta_r < \|r\|_2$, then $\hat{u}$ is well defined and

$$\|\hat{u} - u\|_2 \leq \frac{2\delta_r}{\|r\|_2}, \qquad \langle \hat{u}, u\rangle \geq 1 - 2\frac{\delta_r^2}{\|r\|_2^2}.$$

Hence approximate admissibility turns into the positive-alignment condition needed in Proposition 2.6 whenever $\delta_r < \|r\|_2/\sqrt{2}$. Exact latent SCM identification is unnecessary; sufficient alignment with the true nuisance normal already suffices.

Proof. The first claim is the standard normalization perturbation bound

$$\left\| \frac{a}{\|a\|_2} - \frac{b}{\|b\|_2} \right\|_2 \leq \frac{2\|a - b\|_2}{\|b\|_2}, \qquad a \neq 0,\ b \neq 0,$$

applied with $a = \tilde{r}$ and $b = r$, together with Proposition J.3. The inner-product lower bound follows from

$$\langle \hat{u}, u\rangle = 1 - \frac{1}{2}\|\hat{u} - u\|_2^2.$$

□

Corollary J.5 (Gate-margin robustness)**.** Let $\delta_u \coloneqq 2\delta_r/\|r\|_2$, where $\delta_r = L_s\varepsilon_s + L_t\varepsilon_t$. If the ideal harmful-alignment margin satisfies

$$\langle g, u\rangle \leq -(\tau + m)\|g\|_2$$

for some $m > 0$ and $\delta_u \leq m$, then the practical gate still fires:

$$\langle g, \hat{u}\rangle \leq -\tau\,\|g\|_2.$$

Thus the practical threshold τ is a robustness margin against probe-normal error.

Proof. By Cauchy–Schwarz and Corollary J.4,

$$\langle g, \hat{u}\rangle = \langle g, u\rangle + \langle g, \hat{u} - u\rangle \leq -(\tau + m)\|g\|_2 + \|g\|_2\|\hat{u} - u\|_2 \leq -\tau\|g\|_2.$$

□

Proposition J.6 (Projected-step perturbation under normal error)**.** Suppose the ideal and practical hard-projection branches are both active, and define

$$\Delta^\star(u) \coloneqq -\eta\big(g - \langle g, u\rangle u\big), \qquad \Delta^\star(\hat{u}) \coloneqq -\eta\big(g - \langle g, \hat{u}\rangle\hat{u}\big).$$

Then

$$\|\Delta^\star(\hat{u}) - \Delta^\star(u)\|_2 \leq 2\eta\|g\|_2\,\|\hat{u} - u\|_2 \leq \frac{4\eta\|g\|_2}{\|r\|_2}\,(L_s\varepsilon_s + L_t\varepsilon_t).$$

If the margin condition of Corollary J.5 holds, then the practical hard projection is active whenever the ideal one is, so the same bound compares the implemented hard update to the ideal projected step.

Proof. Using the active-branch closed form and adding and subtracting $\langle g, \hat{u}\rangle u$,

$$\begin{aligned}\|\Delta^\star(\hat{u}) - \Delta^\star(u)\|_2 &= \eta\big\|\langle g, \hat{u}\rangle\hat{u} - \langle g, u\rangle u\big\|_2 \\ &\le \eta\Big(\big|\langle g, \hat{u}\rangle\big|\,\|\hat{u} - u\|_2 + \big|\langle g, \hat{u} - u\rangle\big|\,\|u\|_2\Big) \\ &\le 2\eta\|g\|_2\,\|\hat{u} - u\|_2.\end{aligned}$$

The second inequality follows by substituting the bound from Corollary J.4. □

K Positive alignment improves the true nuisance certificate

We now record the exact first-order identity showing why the approximate-normal guarantees above are sufficient. Corollary J.4 turns admissibility into the positive-alignment premise used below, while Corollary J.5 explains why a cosine threshold $\tau_{\cos}$, and the raw dot-score threshold τ_{dp} used in the released code, can be interpreted as margin-protected harmful-alignment tests at their respective scales.

Let r denote the true cone normal from the nuisance certificate J, and let $\tilde{r}$ be the estimated direction obtained from the intervention probe. Define the unit vector $\hat{u} := \tilde{r}/\|\tilde{r}\|_2$, let

$$a := \langle g, \hat{u}\rangle, \qquad I_- := \mathbb{I}\{a < 0\},$$

and consider

$$\Delta\theta_{\mathrm{ERM}} := -\eta g, \qquad \Delta\theta_{\mathrm{CGS}} := -\eta\big(g - \gamma I_- a\hat{u}\big).$$

Then

$$\Delta\theta_{\mathrm{CGS}} - \Delta\theta_{\mathrm{ERM}} = \eta\gamma I_- a\,\hat{u}. \tag{23}$$

Taking inner products with the true normal r yields

$$\langle r, \Delta\theta_{\mathrm{CGS}} - \Delta\theta_{\mathrm{ERM}}\rangle = \eta\gamma I_- a\,\langle r, \hat{u}\rangle. \tag{24}$$

If $\langle r, \hat{u}\rangle > 0$ and $a < 0$, then the right-hand side of (24) is strictly negative. Therefore the practical CGS correction moves in a direction that is more favorable to the true nuisance certificate than the ERM step.

To make the first-order comparison explicit, use differentiability of J at θ:

$$J(\theta + \Delta) = J(\theta) + \langle r, \Delta\rangle + o(\|\Delta\|).$$

Applying this expansion to $\Delta\theta_{\mathrm{CGS}}$ and $\Delta\theta_{\mathrm{ERM}}$ and subtracting gives

$$J(\theta + \Delta\theta_{\mathrm{CGS}}) - J(\theta + \Delta\theta_{\mathrm{ERM}}) = \eta\gamma I_- a\,\langle r, \hat{u}\rangle + o\big(\|\Delta\theta_{\mathrm{CGS}}\| + \|\Delta\theta_{\mathrm{ERM}}\|\big). \tag{25}$$

Hence, whenever the estimated normal has positive alignment with the true normal and the gate is active, practical CGS decreases the true nuisance certificate more than ERM does to first order. Corollary J.4 provides a concrete sufficient condition for this sign-correct alignment, and Corollary J.5 shows when the same estimated normal also preserves the practical gate decision.

L Causal Target/OOD Generalization Bound

For a source population P and a target population Q over (S, T, Y), define the average causal nuisance radius and semantic-label shift by

$$\rho_Q := \mathbb{E}_{(s,y)\sim P_{S,Y}}\Big[W_{d_T}\big(Q_{T|s,y}, P_{T|s,y}\big)\Big], \qquad \Delta_{SY}(Q, P) := \mathrm{TV}\big(Q_{S,Y}, P_{S,Y}\big). \tag{26}$$

When the target differs only through nuisance interventions, $\Delta_{SY}(Q, P) = 0$ and ρ_Q is the average nuisance transport needed to move the source nuisance mechanism to the target one.

For $\Lambda > 0$, define the restricted empirical nuisance-DRO risk

$$\widehat{\mathcal{R}}_{\rho,\Lambda}(\theta) := \min_{0\le\lambda\le\Lambda}\left\{\lambda\rho + \frac{1}{n}\sum_{i=1}^{n}\sup_{t'\in\mathcal{T}}\big(\ell(\theta; s_i, t', y_i) - \lambda d_T(t', t_i)\big)\right\}. \tag{27}$$

Let

$$\Phi_\Lambda := \left\{ (s,t,y) \mapsto \sup_{t' \in \mathcal{T}} \big(\ell(\theta; s, t', y) - \lambda d_T(t', t) \big) : \theta \in \Theta, 0 \le \lambda \le \Lambda \right\}$$

and write

$$\epsilon_n(\Theta, \Lambda, \delta) := 2\widehat{\mathfrak{R}}_n(\Phi_\Lambda) + 3B\sqrt{\frac{\log(2/\delta)}{2n}},$$

where $\widehat{\mathfrak{R}}_n$ is the empirical Rademacher complexity and $0 \le \ell \le B$.

Theorem L.1 (Causal target/OOD loss bound for CGS)**.** Assume the causal task mechanism is invariant across environments and target shifts act on the nuisance conditional $T \mid S, Y$. Suppose $0 \le \ell \le B$ and the regularity assumptions of Appendix B hold along the training path. Then, with probability at least $1 - \delta$ over the source sample, every $\theta \in \Theta$ and every causally compatible target Q satisfy

$$\mathcal{R}_Q(\theta) := \mathbb{E}_Q[\ell(\theta; S, T, Y)] \le \widehat{\mathcal{R}}_{\rho_Q, \Lambda}(\theta) + \epsilon_n(\Theta, \Lambda, \delta) + B\Delta_{SY}(Q, P). \tag{28}$$

Moreover, let $\theta_{m+1} = \theta_m + \Delta_m$ be the norm-rescaled soft/gated CGS updates for $m = 0, \dots, T-1$, with

$$v_m = g_m - \gamma_m I_m a_m \hat{u}_m, \qquad \Delta_m = -\eta_m \frac{\|g_m\|_2}{\|v_m\|_2} v_m,$$
$$a_m = \langle g_m, \hat{u}_m \rangle, \qquad I_m = \mathbb{I}\{a_m < 0\}.$$

and $q_m = I_m a_m^2 / \|g_m\|_2^2$. Define

$$\chi_\gamma(q) := \frac{1 - \gamma q}{\sqrt{1 - \gamma(2-\gamma)q}}, \qquad \zeta_m = [\Gamma_{\theta_m}(\Delta_m)]_+, \qquad \omega_m = \omega_{\theta_m}(\eta_m \|g_m\|_2).$$

Then the final target risk obeys

$$\begin{aligned} \mathcal{R}_Q(\theta_T) \le \widehat{\mathcal{R}}_{\rho_Q, \Lambda}(\theta_0) + \epsilon_n(\Theta, \Lambda, \delta) + B\Delta_{SY}(Q, P) \\ - \sum_{m=0}^{T-1} \eta_m \|g_m\|_2^2 \chi_{\gamma_m}(q_m) \\ + \sum_{m=0}^{T-1} \Big(\zeta_m + M\varepsilon_{\mathrm{cov}} \eta_m \|g_m\|_2 + \frac{\beta}{2} \eta_m^2 \|g_m\|_2^2 \\ + \omega_m \eta_m \|g_m\|_2 \Big). \end{aligned} \tag{29}$$

Corollary L.2 (Probe admissibility controls the target-risk surcharge)**.** At a training step, suppose the bank surcharge is K_Γ-Lipschitz in the update argument and the ideal hard-projection step using the true cone normal $u = r/\|r\|_2$ is bank-feasible, i.e., $\Gamma_\theta(\Delta^\star(u)) \le 0$. If the practical probe is $(\varepsilon_s, \varepsilon_t)$-admissible and $\delta_r = L_s\varepsilon_s + L_t\varepsilon_t < \|r\|_2$, then the hard-projection step using the probe normal $\hat{u}$ satisfies

$$[\Gamma_\theta(\Delta^\star(\hat{u}))]_+ \le \frac{4K_\Gamma \eta \|g\|_2}{\|r\|_2} \big(L_s\varepsilon_s + L_t\varepsilon_t \big). \tag{30}$$

This section proves Theorem L.1 and Corollary L.2. The proof has three parts. First, a causal nuisance intervention is shown to induce a finite nuisance-only transport radius. Second, nuisance-only DRO is converted into a high-probability empirical target-risk bound. Third, the local CGS majorizer is telescoped along the training path, making the target-risk contribution of cone violation and probe inadmissibility explicit.

Causal target family. Let P denote the source population over (S, T, Y) and let Q denote a target population. The causal compatibility assumption used in the main text is that the target does not alter the task-defining semantic mechanism: labels remain functions of the semantic component in the same way as under P, while the nuisance mechanism $T \mid S, Y$ may change. To isolate nuisance shift from possible class/semantic imbalance, define the hybrid target

$$\overline{Q}(ds, dy, dt) := P_{S,Y}(ds, dy)\, Q(dt \mid s, y). \tag{31}$$

Thus $\overline{Q}$ has the source semantic-label marginal and the target nuisance conditional. The quantities in Eq. (26) are

$$\rho_Q = \mathbb{E}_{(s,y)\sim P_{S,Y}}\left[W_{d_T}\big(Q_{T|s,y}, P_{T|s,y}\big)\right], \qquad \Delta_{SY}(Q,P) = \mathrm{TV}(Q_{S,Y}, P_{S,Y}).$$

When target environments are pure nuisance interventions, $Q_{S,Y} = P_{S,Y}$ and hence $Q = \overline{Q}$.

Lemma L.3 (Causal nuisance interventions induce nuisance-only transport)**.** For the nuisance-only cost c in Eq. (5), the hybrid target satisfies

$$W_c(\overline{Q}, P) \le \rho_Q. \tag{32}$$

Proof. Because $\overline{Q}$ and P share the same (S,Y) marginal, we may couple (S,Y) identically. For each realized pair (s,y), choose an optimal, or ϵ-optimal, coupling $\pi_{s,y}$ between $P_{T|s,y}$ and $Q_{T|s,y}$ under the metric d_T. Combining the identity coupling on (S,Y) with $\pi_{s,y}$ gives a valid coupling between P and $\overline{Q}$. Under this coupling the semantic coordinate is unchanged, so the cost is finite and equals $d_T(T,T')$. Therefore the expected transport cost is

$$\mathbb{E}_{(s,y)\sim P_{S,Y}}\mathbb{E}_{(T,T')\sim\pi_{s,y}} d_T(T,T') \le \mathbb{E}_{(s,y)\sim P_{S,Y}}\left[W_{d_T}\big(Q_{T|s,y}, P_{T|s,y}\big)\right] + \epsilon.$$

Letting $\epsilon \downarrow 0$ yields (32). □

Lemma L.4 (Population target risk is dominated by nuisance-only DRO)**.** Assume $0 \le \ell \le B$. Then every parameter vector θ satisfies

$$\mathcal{R}_Q(\theta) \le \mathcal{R}^P_{\rho_Q}(\theta) + B\Delta_{SY}(Q,P), \tag{33}$$

where $\mathcal{R}^P_\rho(\theta) := \sup_{Q':W_c(Q',P)\le\rho} \mathbb{E}_{Q'}[\ell(\theta;S,T,Y)]$ is the population nuisance-only DRO risk around P.

Proof. First compare Q to the hybrid target $\overline{Q}$. Since Q and $\overline{Q}$ share the same conditional $T \mid S,Y$ and $0 \le \ell \le B$,

$$\mathbb{E}_Q[\ell(\theta;S,T,Y)] \le \mathbb{E}_{\overline{Q}}[\ell(\theta;S,T,Y)] + B\,\mathrm{TV}(Q_{S,Y}, P_{S,Y}).$$

By Lemma L.3, $\overline{Q}$ lies inside the nuisance-only Wasserstein ball of radius ρ_Q around P. Therefore

$$\mathbb{E}_{\overline{Q}}[\ell(\theta;S,T,Y)] \le \sup_{Q':W_c(Q',P)\le\rho_Q} \mathbb{E}_{Q'}[\ell(\theta;S,T,Y)] = \mathcal{R}^P_{\rho_Q}(\theta).$$

Combining the two displays proves (33). □

Uniform empirical dual control. For $\Lambda > 0$, define the restricted population and empirical dual risks

$$\begin{aligned}
\mathcal{R}^P_{\rho,\Lambda}(\theta) &:= \min_{0\le\lambda\le\Lambda}\left\{\lambda\rho + \mathbb{E}_{(s,t,y)\sim P}\phi_{\theta,\lambda}(s,t,y)\right\}, \\
\widehat{\mathcal{R}}_{\rho,\Lambda}(\theta) &:= \min_{0\le\lambda\le\Lambda}\left\{\lambda\rho + \frac{1}{n}\sum_{i=1}^n \phi_{\theta,\lambda}(s_i,t_i,y_i)\right\},
\end{aligned} \tag{34}$$

where

$$\phi_{\theta,\lambda}(s,t,y) := \sup_{t'\in\mathcal{T}}\big(\ell(\theta;s,t',y) - \lambda d_T(t',t)\big).$$

Since $d_T \ge 0$ and $0 \le \ell \le B$, the dual loss class $\Phi_\Lambda = \{\phi_{\theta,\lambda} : \theta\in\Theta, 0\le\lambda\le\Lambda\}$ is bounded in $[0,B]$.

Lemma L.5 (Empirical restricted DRO upper-bounds target risk)**.** With probability at least $1-\delta$ over the source sample, simultaneously for all $\theta\in\Theta$, all $\rho\ge 0$, and all causally compatible targets Q,

$$\mathcal{R}_Q(\theta) \le \widehat{\mathcal{R}}_{\rho_Q,\Lambda}(\theta) + 2\widehat{\mathfrak{R}}_n(\Phi_\Lambda) + 3B\sqrt{\frac{\log(2/\delta)}{2n}} + B\Delta_{SY}(Q,P). \tag{35}$$

Proof. By standard symmetrization together with the additional McDiarmid concentration step that converts the expected Rademacher complexity into the empirical Rademacher complexity, with probability at least $1-\delta$,

$$\sup_{\theta\in\Theta,\,0\le\lambda\le\Lambda}\left|\mathbb{E}_P\phi_{\theta,\lambda}-\frac{1}{n}\sum_{i=1}^{n}\phi_{\theta,\lambda}(s_i,t_i,y_i)\right|\le 2\widehat{\mathfrak{R}}_n(\Phi_\Lambda)+3B\sqrt{\frac{\log(2/\delta)}{2n}}. \tag{36}$$

On this event, for any fixed θ and ρ,

$$\begin{aligned}\mathcal{R}_\rho^P(\theta)&\le\mathcal{R}_{\rho,\Lambda}^P(\theta)\\&=\min_{0\le\lambda\le\Lambda}\{\lambda\rho+\mathbb{E}_P\phi_{\theta,\lambda}\}\\&\le\min_{0\le\lambda\le\Lambda}\left\{\lambda\rho+\frac{1}{n}\sum_{i=1}^{n}\phi_{\theta,\lambda}(s_i,t_i,y_i)\right\}+2\widehat{\mathfrak{R}}_n(\Phi_\Lambda)+3B\sqrt{\frac{\log(2/\delta)}{2n}}\\&=\widehat{\mathcal{R}}_{\rho,\Lambda}(\theta)+2\widehat{\mathfrak{R}}_n(\Phi_\Lambda)+3B\sqrt{\frac{\log(2/\delta)}{2n}}.\end{aligned}$$

The first inequality uses the fact that restricting the dual minimization to $[0,\Lambda]$ can only increase the DRO value and hence remains a valid upper bound. Substituting $\rho=\rho_Q$ and then applying Lemma L.4 proves (35). □

Telescoping the CGS local robust-risk changes. It remains to show how CGS changes the empirical robust term in (35). Let

$$\widehat{F}_{\rho,\Lambda}(\theta,\lambda):=\lambda\rho+\frac{1}{n}\sum_{i=1}^{n}\sup_{t'\in\mathcal{T}}\big(\ell(\theta;s_i,t',y_i)-\lambda d_T(t',t_i)\big),\qquad 0\le\lambda\le\Lambda.$$

For each iterate θ_m, choose $\lambda_m\in\arg\min_{0\le\lambda\le\Lambda}\widehat{F}_{\rho_Q,\Lambda}(\theta_m,\lambda)$. The same local argument as Theorem 2.1, applied to the empirical objective with fixed multiplier λ_m, gives for sufficiently small Δ_m,

$$\widehat{\mathcal{R}}_{\rho_Q,\Lambda}(\theta_{m+1})\le\widehat{\mathcal{R}}_{\rho_Q,\Lambda}(\theta_m)+\langle g_m,\Delta_m\rangle+\big[\Gamma_{\theta_m}(\Delta_m)\big]_+ +M\varepsilon_{\mathrm{cov}}\|\Delta_m\|_2+\frac{\beta}{2}\|\Delta_m\|_2^2. \tag{37}$$

Indeed, $\widehat{\mathcal{R}}_{\rho_Q,\Lambda}(\theta_{m+1})\le\widehat{F}_{\rho_Q,\Lambda}(\theta_{m+1},\lambda_m)$ by optimality at the new point, while $\widehat{F}_{\rho_Q,\Lambda}(\theta_m,\lambda_m)=\widehat{\mathcal{R}}_{\rho_Q,\Lambda}(\theta_m)$ by the choice of λ_m. The local majorizer supplies the remaining terms.

Summing (37) from $m=0$ to $T-1$ yields

$$\widehat{\mathcal{R}}_{\rho_Q,\Lambda}(\theta_T)\le\widehat{\mathcal{R}}_{\rho_Q,\Lambda}(\theta_0)+\sum_{m=0}^{T-1}\left(\langle g_m,\Delta_m\rangle+\zeta_m+M\varepsilon_{\mathrm{cov}}\|\Delta_m\|_2+\frac{\beta}{2}\|\Delta_m\|_2^2\right), \tag{38}$$

where $\zeta_m=[\Gamma_{\theta_m}(\Delta_m)]_+$. For the norm-rescaled soft/gated CGS step,

$$\Delta_m=-\eta_m\frac{\|g_m\|_2}{\|v_m\|_2}v_m,\qquad v_m=g_m-\gamma_m I_m a_m\hat{u}_m,$$

so $\|\Delta_m\|_2=\eta_m\|g_m\|_2$. The same calculation used in Appendix M gives

$$\langle g_m,\Delta_m\rangle=-\eta_m\|g_m\|_2^2\frac{1-\gamma_m q_m}{\sqrt{1-\gamma_m(2-\gamma_m)q_m}}=-\eta_m\|g_m\|_2^2\chi_{\gamma_m}(q_m). \tag{39}$$

Substituting (39) and $\|\Delta_m\|_2=\eta_m\|g_m\|_2$ into (38), then applying Lemma L.5 to θ_T, proves Eq. (29). The static bound (28) is the special case obtained by applying Lemma L.5 directly to a single θ.

Probe-admissibility surcharge. We now prove Corollary L.2. Assume Γ_θ is K_Γ-Lipschitz in its update argument:

$$|\Gamma_\theta(\Delta)-\Gamma_\theta(\Delta')|\le K_\Gamma\|\Delta-\Delta'\|_2.$$

If the ideal hard-projection step is bank-feasible, then $[\Gamma_\theta(\Delta^\star(u))]_+ = 0$. Hence

$$\begin{aligned}\big[\Gamma_\theta(\Delta^\star(\hat{u}))\big]_+ &\le \big[\Gamma_\theta(\Delta^\star(u))\big]_+ + K_\Gamma\|\Delta^\star(\hat{u}) - \Delta^\star(u)\|_2 \\ &\le K_\Gamma\|\Delta^\star(\hat{u}) - \Delta^\star(u)\|_2.\end{aligned}$$

By Proposition J.6, under the active hard-projection branch,

$$\|\Delta^\star(\hat{u}) - \Delta^\star(u)\|_2 \le \frac{4\eta\|g\|_2}{\|r\|_2}\big(L_s\varepsilon_s + L_t\varepsilon_t\big).$$

Combining the last two displays proves Eq. (30). This completes the bridge from approximate causal admissibility to an explicit term in the target-risk bound.

M One-step decrease of the Norm-Rescaled Practical Cone Update

We establish a sufficient decrease bound for a single iteration of the norm-rescaled practical cone update under standard smoothness. Let $\Delta R_\nu : \mathbb{R}^d \to \mathbb{R}$ be differentiable and L_θ-smooth in parameter space, meaning that for all $x, y \in \mathbb{R}^d$,

$$\Delta R_\nu(y) \le \Delta R_\nu(x) + \langle \nabla \Delta R_\nu(x), y - x\rangle + \frac{L_\theta}{2}\,\|y - x\|_2^2. \tag{40}$$

Fix a parameter vector $\theta \in \mathbb{R}^d$ and define $g \coloneqq \nabla_\theta \Delta R_\nu(\theta)$. Let $\hat{u}$ denote the estimated unit cone normal and define

$$a \coloneqq \langle g, \hat{u}\rangle, \qquad I_- \coloneqq \mathbb{I}\{a < 0\}, \qquad v \coloneqq g - \gamma I_- a\hat{u}.$$

For a step size $\eta > 0$, the norm-rescaled practical update is

$$\Delta\theta^\star \;=\; -\,\eta\,\frac{\|g\|_2}{\|v\|_2}\,v. \tag{41}$$

We write

$$q \coloneqq \frac{I_- a^2}{\|g\|_2^2} \in [0, 1].$$

We now prove Theorem C.1.

Proof of Theorem C.1. Apply the smoothness inequality (40) with $x = \theta$ and $y = \theta + \Delta\theta^\star$:

$$\Delta R_\nu(\theta + \Delta\theta^\star) \le \Delta R_\nu(\theta) + \langle g, \Delta\theta^\star\rangle + \frac{L_\theta}{2}\|\Delta\theta^\star\|_2^2. \tag{42}$$

Since $\|\Delta\theta^\star\|_2 = \eta\|g\|_2$, the quadratic term is exactly $\frac{L_\theta\eta^2}{2}\|g\|_2^2$. For the linear term,

$$\langle g, \Delta\theta^\star\rangle = -\eta\frac{\|g\|_2}{\|v\|_2}\langle g, v\rangle.$$

Now

$$\langle g, v\rangle = \|g\|_2^2 - \gamma I_- a^2 = \|g\|_2^2(1 - \gamma q),$$

and, using $\|\hat{u}\|_2 = 1$,

$$\|v\|_2^2 = \|g - \gamma I_- a\hat{u}\|_2^2 = \|g\|_2^2 - \gamma(2-\gamma)I_- a^2 = \|g\|_2^2\big(1 - \gamma(2-\gamma)q\big).$$

Hence

$$\langle g, \Delta\theta^\star\rangle = -\eta\|g\|_2^2\,\frac{1 - \gamma q}{\sqrt{1 - \gamma(2-\gamma)q}}.$$

Substituting this and the quadratic term into (42) yields (43).

To obtain a uniform lower bound on the ratio, define

$$r_\gamma(q) \coloneqq \frac{1 - \gamma q}{\sqrt{1 - \gamma(2-\gamma)q}}, \qquad q \in [0, 1].$$

A direct calculus computation shows that r_γ attains its minimum at $q^\star = \frac{1}{2-\gamma}$ with value $r_\gamma(q^\star) = m_\gamma = \frac{2\sqrt{1-\gamma}}{2-\gamma}$. Therefore $r_\gamma(q) \geq m_\gamma$ on $[0,1]$, which combined with (15) gives

$$\Delta R_\nu(\theta + \Delta\theta^\star) \leq \Delta R_\nu(\theta) - \eta\Big(m_\gamma - \tfrac{L_\theta \eta}{2}\Big)\|g\|_2^2. \tag{43}$$

If $\eta \leq m_\gamma / L_\theta$, then $m_\gamma - \frac{L_\theta \eta}{2} \geq \frac{m_\gamma}{2}$, yielding

$$\Delta R_\nu(\theta + \Delta\theta^\star) \leq \Delta R_\nu(\theta) - \frac{\eta\, m_\gamma}{2}\|g\|_2^2,$$

which is the second conclusion of Theorem C.1. □

N Proof of Curvature Suppression

The curvature proposition concerns the active *hard-projection* geometry. In that case the feasible boundary is the tangent subspace orthogonal to the cone normal, and the maximal Rayleigh quotient on that subspace admits an exact upper bound.

Setup. Let $H(\theta)$ be a symmetric Hessian with eigenpairs (λ_i, v_i) ordered as $\lambda_1 \geq \lambda_2 \geq \cdots$, and let u be a unit cone normal. We study the tangent subspace

$$S \coloneqq \{v \in \mathbb{R}^d : \langle v, u\rangle = 0\}.$$

Write

$$u = \sum_{i=1}^d c_i v_i, \qquad v = \sum_{i=1}^d \alpha_i v_i$$

for any unit vector $v \in S$, so $\sum_i c_i^2 = \sum_i \alpha_i^2 = 1$ and $0 = \langle u, v\rangle = \sum_i c_i \alpha_i$.

Rayleigh quotient on the tangent subspace. For any unit $v \in S$,

$$v^\top H(\theta) v = \sum_{i=1}^d \lambda_i \alpha_i^2 \leq \lambda_1 \alpha_1^2 + \lambda_2 \sum_{i=2}^d \alpha_i^2 = \lambda_2 + (\lambda_1 - \lambda_2)\alpha_1^2.$$

It therefore remains to bound α_1^2 for vectors orthogonal to u.

From $0 = \sum_i c_i \alpha_i$, we have

$$|c_1 \alpha_1| = \left|\sum_{i=2}^d c_i \alpha_i\right| \leq \Big(\sum_{i=2}^d c_i^2\Big)^{1/2} \Big(\sum_{i=2}^d \alpha_i^2\Big)^{1/2} = \sqrt{1 - c_1^2}\sqrt{1 - \alpha_1^2},$$

where we used Cauchy–Schwarz. Squaring both sides gives

$$c_1^2 \alpha_1^2 \leq (1 - c_1^2)(1 - \alpha_1^2),$$

which rearranges to

$$\alpha_1^2 \leq 1 - c_1^2 = 1 - \langle u, v_1\rangle^2.$$

Substituting this into the Rayleigh quotient bound yields

$$v^\top H(\theta) v \leq \lambda_2 + (\lambda_1 - \lambda_2)\big(1 - \langle u, v_1\rangle^2\big) = \lambda_1 - (\lambda_1 - \lambda_2)\langle u, v_1\rangle^2.$$

Taking the maximum over all unit $v \in S$ proves Proposition C.2.

O Detailed Per-Domain Benchmark Results

Main DG protocol and implementation. We evaluate on PACS [60], VLCS [61], OfficeHome [62], TerraIncognita [1], and DomainNet [63]. For all five datasets we follow the standard leave-one-domain-out DomainBed protocol [64]: train on all but one source domain, select hyperparameters with the DomainBed training-domain validation split, test on the held-out domain, and report the average target-domain classification accuracy. We use DomainBed default preprocessing and training

Algorithm 2: CGSFactory: Drop-in Wrapper for DomainBed Algorithms

Require: Base algorithm $\mathcal{A}$ exposing a gradient-computation routine; frequency k
Initialize $\mathcal{A}$; set counter $t \leftarrow 0$
function CGSFactory.Update($\mathcal{B}$)
  $\ell, g \leftarrow \mathcal{A}.\textsc{ComputeGradient}(\mathcal{B})$ // No optimizer step yet
  if $t \bmod k = 0$ **then**
    $g \leftarrow \textsc{CGSSteer}(g)$ // Apply CGS as in Algorithm 1
  $\textsc{Step}(g)$
  $t \leftarrow t + 1$
  return ℓ

augmentations for the ResNet-50 benchmark setting, an ImageNet-pretrained ResNet-50 backbone [65], Adam, learning rate 5×10^{-5}, and batch size 32. Unless otherwise stated, CGS uses center patch shuffling as the intervention probe with patch size 2, update period $k = 100$, projection strength $\gamma = 0.5$, and raw dot-score gate threshold $\tau_{\mathrm{dp}} = 0.2$. Because CGS applies its steering correction only every k steps, we smooth the model weights using a simple moving average (SMA) for evaluation [49], as in prior work [52]; this stabilization changes only the evaluation model and never turns the destroyed view into an auxiliary supervised example.

Reproducibility details. The main classification runs can be reproduced with the DomainBed-style command

```
python -m domainbed.scripts.train
-data_dir DATA_ROOT -dataset DATASET
-algorithm ERM_GRAD_INHIB_FAST_conditional_normalized2_dp_thresh_
layer
-test_envs ENV_ID -hparams CGS_HPARAMS_JSON
-hparams_seed 0 -trial_seed 0
```

For the locked CGS configuration, `CGS_HPARAMS_JSON` sets `shuffle_patch_size=2`, `shuffle_patch_every=100`, `shuffle_fn=shuffle_center_patches`, `grad_inhib_strength=0.5`, `batch_size=32`, `weight_decay=0.0003`, `lr=5e-5`, `resnet_dropout=0`, and `dot_prod_thresh=0.2`. The classification logs use `holdout_fraction=0.2`, `uda_holdout_fraction=0`, `hparams_seed=0`, `trial_seed=0`, and a recorded per-run random seed. The software environment for the reported classification runs was Python 3.10.18, PyTorch 2.6.0+cu124, Torchvision 0.21.0+cu124, CUDA 12.4, cuDNN 90100, NumPy 2.1.2, and PIL 11.0.0. The machine used NVIDIA RTX A6000 GPUs with 48GB memory (driver 535.183.01), dual Intel Xeon Gold 6226R CPUs (64 logical CPUs total), and 187 GiB system RAM.

Table 4: Main results on standard domain generalization benchmarks (PACS, VLCS, OfficeHome, TerraIncognita, DomainNet). We report the average classification accuracy (%) using the leave-one-domain-out protocol. The best result in each column is in **bold**, and the best average result is also in **bold**. Results for baseline methods are taken from their original publications or established benchmark reports; matched wrapper comparisons are reported separately in Appendix P.

Model	PACS					VLCS					OfficeHome					TerraIncognita					DomainNet						
	A	C	P	S	Avg	C	L	S	V	Avg	A	C	P	R	Avg	L100	L38	L43	L46	Avg	clip	info	paint	quick	real	sketch	Avg
ERM	84.7	80.8	97.2	79.3	85.5	97.7	64.3	73.4	74.6	77.5	61.3	52.4	75.8	76.6	66.5	49.8	42.1	56.9	35.7	46.1	58.6	19.2	47.0	13.2	59.9	49.8	41.3
IRM [4]	84.8	76.4	96.7	76.1	83.5	98.6	64.9	73.4	77.3	78.6	58.9	52.2	72.1	74.0	64.3	54.6	39.8	56.2	39.6	47.6	40.4	12.1	31.4	9.8	37.7	36.7	28.0
GDRO [41]	83.5	79.1	96.7	78.3	84.4	97.3	63.4	69.5	76.7	76.7	60.4	52.7	75.0	76.0	66.0	41.2	38.6	56.7	36.4	43.2	47.2	17.5	34.2	9.2	51.9	40.1	33.4
Mixup [8]	86.1	78.9	97.6	75.8	84.6	98.3	64.8	72.1	74.3	77.4	62.4	54.8	76.9	78.3	68.1	**59.6**	42.2	55.9	33.9	47.9	55.6	18.7	45.1	12.8	57.6	48.2	39.6
MLDG [15]	85.5	80.1	97.4	76.6	84.9	97.4	65.2	71.0	75.3	77.2	61.5	53.2	75.0	77.5	66.8	54.2	44.3	55.6	36.9	47.7	59.3	19.6	46.8	13.4	60.1	50.4	41.6
CORAL [10]	88.3	80.0	97.5	78.8	86.2	98.3	**66.1**	73.4	77.5	78.8	65.3	54.4	76.5	78.4	68.7	51.6	42.2	57.0	39.8	47.6	59.2	19.9	47.4	14.0	59.8	50.4	41.8
MMD [12]	86.1	79.4	96.6	76.5	84.6	97.7	64.0	72.8	75.3	77.5	60.4	53.3	74.3	77.4	66.3	41.9	34.8	57.0	35.2	42.2	32.2	11.2	26.8	8.8	32.7	29.0	23.5
DANN [11]	86.4	77.4	97.3	73.5	83.6	**98.8**	65.1	73.1	77.2	78.6	59.9	53.0	73.6	76.9	65.9	51.1	40.6	57.4	37.7	46.7	53.1	18.3	44.2	11.9	55.5	46.8	38.3
CDANN [42]	84.6	75.5	96.8	73.5	82.6	97.1	65.1	70.7	77.1	77.5	61.5	50.4	74.4	76.6	65.8	47.0	41.3	54.9	39.8	45.8	54.6	17.3	44.2	12.8	56.2	45.9	38.5
MTL [43]	87.5	77.1	96.4	77.3	84.6	97.8	64.3	71.5	75.3	77.2	61.5	52.4	74.9	76.8	66.4	49.3	39.6	55.6	37.8	45.6	58.0	19.2	46.2	12.7	59.9	49.0	40.8
SagNet [44]	87.4	80.7	97.1	80.0	86.3	97.9	64.5	71.4	77.5	77.8	63.4	54.8	75.8	78.3	68.1	53.0	43.0	57.9	40.4	48.6	57.7	19.1	46.3	13.5	58.9	49.5	40.8
ARM [45]	86.8	76.8	97.4	79.3	85.1	98.7	63.6	71.3	76.7	77.6	58.9	51.0	74.1	75.2	64.8	49.3	38.3	55.8	38.7	45.5	49.6	16.5	41.5	10.8	53.5	43.9	36.0
VREx [14]	86.0	79.1	96.9	77.7	84.9	98.4	64.4	74.1	76.2	78.3	60.7	53.0	75.3	76.6	66.4	48.2	41.7	56.8	38.7	46.4	43.3	14.1	32.5	9.8	43.5	37.7	30.1
RSC [46]	85.4	79.7	97.6	78.2	85.2	97.9	62.5	72.3	75.6	77.1	60.7	51.4	74.8	75.1	65.5	50.2	39.2	56.3	40.8	46.6	55.0	18.3	44.4	12.5	55.7	47.8	38.9
SWAD [48]	89.3	83.4	97.3	82.5	88.1	**98.8**	63.3	75.3	79.2	79.1	66.1	57.7	78.4	80.2	70.6	55.4	44.9	59.7	39.9	50.0	66.0	22.4	53.5	16.1	65.8	55.5	46.5
PCL [50]	**90.2**	83.9	98.1	82.6	88.7	-	-	-	-	-	67.3	59.9	78.7	80.7	71.6	58.7	46.3	60.0	43.6	52.1	67.9	24.3	55.3	15.7	66.6	56.4	47.7
INSURE [52]	**90.2**	**85.3**	97.9	**83.8**	89.3	**98.8**	63.8	72.2	**81.4**	79.1	**71.4**	57.3	78.0	81.2	72.0	58.8	46.4	61.7	**45.5**	**53.1**	67.8	24.0	55.6	16.0	67.6	57.2	48.0
CGS (Ours)	89.6	84.1	**99.0**	**83.8**	89.1	98.7	64.7	**75.5**	79.2	**79.5**	69.6	**61.0**	**80.5**	**83.2**	**73.6**	54.0	**50.3**	**62.1**	44.3	52.7	**69.9**	**25.9**	**57.8**	**17.3**	**69.6**	**59.9**	**50.1**

Table 5: Applying CGS as an optimizer-side wrapper to existing DG methods. For each method, both the unwrapped baseline and CGS-wrapped variant are run on the same machine using the default DomainBed implementation and the author-provided/default hyperparameters. CGS uses one fixed canonical hyperparameter configuration across all dataset–algorithm combinations. Parenthesized values are matched-run gains over the corresponding unwrapped baseline. Averages are computed over the datasets on which wrapper runs were conducted.

Algorithm	**PACS**	**VLCS**	**OfficeHome**	**TerraInc**	**4-dataset Avg**
MixStyle[9]	88.1 (+3.7)	78.8 (+2.0)	69.7 (+3.5)	49.8 (+4.0)	71.6
GroupDRO [41]	88.2 (+3.8)	77.3 (+1.2)	73.1 (+8.8)	48.7 (+5.5)	71.8
MMD [12]	88.4 (+3.9)	78.2 (+1.6)	73.0 (+8.7)	45.9 (+3.0)	71.4
AND-mask [66]	86.6 (+2.7)	76.7 (+0.7)	72.6 (+2.8)	47.1 (+4.1)	70.8
ARM [45]	87.5 (+3.4)	78.3 (+1.2)	72.9 (+5.8)	49.6 (+4.1)	72.1
IRM [4]	85.8 (+2.3)	79.0 (+1.6)	65.5 (+8.4)	46.6 (-0.2)	69.2
MTL [43]	88.4 (+3.8)	77.9 (+2.8)	73.0 (+7.3)	49.0 (+5.0)	72.1
VREx [14]	87.6 (+2.7)	78.4 (+1.2)	73.1 (+7.0)	52.0 (+5.6)	72.8
MLDG [15]	88.2 (+7.1)	79.0 (+8.4)	71.9 (+15.8)	47.3 (+14.5)	71.6
Mixup [8]	88.4 (+2.0)	78.4 (+1.6)	73.6 (+4.0)	48.9 (+2.5)	72.3
SagNet [44]	88.5 (+5.0)	78.6 (+2.1)	73.4 (+5.9)	52.4 (+5.1)	73.2
CORAL [10]	88.5 (+3.7)	77.5 (-0.1)	73.1 (+3.8)	47.7 (+6.0)	71.7
RSC [46]	87.6 (+4.2)	80.3 (+3.0)	72.6 (+7.2)	46.2 (-0.4)	71.7
Fish [16]	88.1 (+2.6)	77.8 (+0.3)	73.2 (+7.2)	47.1(+3.7)	71.6
SAM [67]	87.6 (+1.0)	79.2 (+1.7)	73.3 (+2.9)	51.0 (+3.5)	72.8
GSAM [68]	87.8 (+1.1)	79.3 (+0.3)	73.6 (+2.7)	50.5 (+11.6)	72.8
SAGM [51]	88.1 (+1.4)	79.0 (+1.4)	73.7 (+2.7)	51.3 (+4.7)	73.0

P Augmenting Existing DG Algorithms with CGS

We further evaluate CGS as a plug-in module to augment existing domain generalization algorithms. The process is shown in Algorithm 2, where we propose CGS as an optimizer-side wrapper. Every k steps, we compute the ERM gradient on the current minibatch and estimate a cone normal from a probe-transformed version of the same batch. The wrapper then projects the per-layer gradient away from this nuisance-aligned direction before the optimizer update step. This process highlights the simplicity of integration: CGS only overrides the update rule in a DomainBed [64]-style implementation framework, with no architectural changes.

In this setting, we use a single fixed set of hyperparameters across all wrapper runs, without dataset-specific tuning. Table 5 reports the four-dataset wrapper average for the datasets on which these runs were conducted. Unlike the reference benchmark table, each parenthesized gain is measured against the corresponding unwrapped baseline in the same DomainBed-style code path and machine setting. The broad pattern is therefore the central evidence for the plug-in claim: the same optimizer-side correction improves a diverse set of baselines (up to +15.8 percentage points) and can be integrated without architectural changes.

Q Comparison with Gradient-Based Methods

We further situate CGS within the family of gradient-based domain generalization algorithms. Table 6 summarizes key differences in domain label requirements and computational cost per training step. Approaches such as Fish [16], Fishr [18], SAND-mask [69], and GGA [54] explicitly rely on domain labels to compute per-domain gradients. This design introduces significant overhead; Fish performs k inner updates per outer step, SAND-mask scales linearly with the number of source domains E, and GGA requires an annealing phase with domain-wise gradients before reverting to ERM-style updates.

Other methods, including SAM [67], GSAM [68], and SAGM [51], avoid explicit domain labels but nevertheless double the compute cost per iteration by requiring two forward and two backward passes. In contrast, CGS attains a favorable balance. It requires only a single forward and backward pass per step, matching the efficiency of ERM, while sparsely inserting an additional forward/backward pair every k iterations to compute the inhibition signal. Averaged over training, the pass count is $2 + 2/k$ versus ERM's 2; for the default $k = 100$, this is a $1.01\times$ pass-count overhead. CGS is used only during training, so inference cost, memory footprint at test time, and the deployed architecture are

Table 6: Comparison of gradient-based domain generalization methods on compute per training step. The last column reports the *total* number of forward + backward passes per optimizer step (#(Fwd–Bkwd)/Step; ↓ means lower is better). **Bold indicates the cheapest variant.** ERM serves as a baseline (not a DG method) and is included only for comparison.

Method	Venue / Year	Domain ID	#(Fwd–Bkwd)/Step ↓
ERM	–	No	2
SAM [67]	ICLR '21	No	4
GSAM [68]	ICLR '22	No	4
SAGM [51]	CVPR '23	No	4
Fish [16]	ICLR '22	Yes	$\approx 2k$
Fishr [18]	ICML '22	Yes	2
SAND-mask [69]	arXiv '21	Yes	$2E$
GGA [54]	CVPR '25	Yes	anneal: $2E$; after: 2
CGS (Ours)	**–**	**No**	$\mathbf{2}$ $(+\mathbf{2}$ every $\mathbf{k}$ steps)

unchanged. This yields a near-ERM training cost, independent of both the number of domains E and the choice of k in inner-loop formulations.

Measured wall-clock example. As a concrete timing check, we compare representative matched classification runs with the locked batch size 32 and learning rate 5×10^{-5}. Excluding the cold step 0 measurement, CGS averages 0.249736 seconds per training step over checkpoints from steps 300–5000, while ERM averages 0.221294 seconds per training step over checkpoints from steps 300–900. This corresponds to a measured $1.129\times$ ERM step time, or 12.9% overhead, and estimates 5000 training steps at 20.8 minutes for CGS versus 18.4 minutes for ERM on the same hardware class. Peak allocated GPU memory in these logs is 8.35 GiB for CGS and 8.14 GiB for ERM. These wall-clock values are implementation- and hardware-dependent, but they support the pass-count analysis: the amortized training cost remains close to ERM, and inference cost is identical because the deployed network is unchanged.

Taken together, these comparisons highlight CGS as an efficient alternative within gradient-based methods, as it retains the benefits of gradient steering while avoiding the prohibitive per-domain computations or doubled forward/backward pass requirements that characterize prior approaches.

R Single Source Domain Generalization Results

We further evaluate our method in the Single Source Domain Generalization (SDG) setting, where only one domain is used for training and the remaining domains are treated as unseen targets. This setup is considerably more challenging than the multi-source scenario (used in Tables 1- 4) due to the lack of cross-domain supervision during training. We maintain all hyperparameters identical to those in our multi-source experiments, thereby avoiding any additional tuning for this setting.

On PACS (Table 7), CGS achieves an average accuracy of 73.1% (+4.2% over the strongest listed prior result), outperforming all listed approaches. Notably, it improves generalization to the Sketch domain, where methods such as Meta-Casual and ITTA show strong but lower performance, highlighting the ability of CGS to leverage causal gradients for robustness to severe domain shifts.

On OfficeHome (Table 8), CGS attains the best overall accuracy among the listed methods, 58.4% (+2.1%), surpassing recent gradient-based methods such as Fishr [18] and SAGM [51]. It is strongest on three of the four target domains, indicating consistent effectiveness on this large-scale and diverse benchmark. Together, these results show that CGS remains effective in both single- and multi-source generalization setups without task-specific hyperparameter adaptation.

Recent single-domain generalization methods such as AdvST [70] strengthen augmentation or adversarial augmentation pipelines. CGS is complementary: the destroyed image is not treated as a label-preserving positive example, but as a negative/probe view whose gradient identifies a shortcut direction. A key factor behind the superior SDG performance of CGS compared to gradient-based baselines lies in how gradients are manipulated. Prior methods regularize or match gradient statistics across domains, which becomes less effective when only a single source is available, as

no inter-domain signal can be exploited. In contrast, CGS explicitly steers gradients away from nuisance-aligned directions by leveraging structured interventions on the feature space, a process that does not depend on having multiple domains. This independence from cross-domain supervision explains why CGS maintains strong generalization in SDG while gradient-matching approaches struggle. When combined with the multi-source results in Section 3.1, the findings suggest that the effectiveness of CGS comes from the same within-sample destruction-and-steering principle rather than from reliance on domain diversity.

Table 7: SDG results (%) on PACS. One domain is used as the source domain and the others are used as the target domain.

Method	**P**	**A**	**C**	**S**	**Avg.**
AugMix [71]	38.30	66.54	70.16	52.48	56.87
RSC [46]	41.60	73.40	75.90	56.20	61.80
L2D [72]	52.29	76.91	77.88	53.66	65.18
RSC+ASR [46]	54.60	76.70	79.30	61.00	67.90
pAdaIn [73]	33.66	64.96	65.24	32.04	49.98
ERM [64]	33.65	65.38	64.06	34.15	49.34
MixStyle [9]	37.44	67.60	70.38	32.58	52.50
EFDMix [36]	42.50	63.20	73.90	38.10	54.40
DSU [74]	42.10	71.54	74.51	47.75	58.97
ACVC [75]	48.05	73.68	77.53	55.67	63.73
MAD [76]	52.95	75.51	77.25	57.95	65.87
P-RC [77]	57.11	76.98	**84.12**	56.29	68.88
ITTA [78]	56.50	78.40	79.60	60.80	68.00
CGS (Ours)	**65.2**	**80.9**	81.2	**65.2**	**73.1**

Table 8: SDG results (%) on OfficeHome. One domain is used as the source domain and the others are used as the target domain.

Method	**A**	**C**	**P**	**R**	**Avg.**
DANN [11]	55.20	49.30	48.40	58.40	52.80
CORAL [10]	56.50	52.80	50.30	59.40	54.50
IRM [4]	54.90	53.20	48.60	59.20	54.00
Mixup [8]	55.50	54.10	49.40	59.20	54.60
GroupDRO [41]	55.10	52.00	50.30	59.30	54.20
MTL [43]	55.30	53.20	49.00	60.20	54.50
VREx [14]	55.50	52.60	49.10	59.30	54.10
MixStyle [9]	44.30	29.80	33.00	48.00	38.80
ERM [64]	55.60	52.80	50.30	59.40	54.50
SAM [67]	56.90	53.80	50.90	59.60	55.30
SagNet [44]	56.90	53.80	50.20	61.20	55.60
Fishr [18]	56.90	55.20	52.50	59.20	55.50
RIDG [79]	56.80	55.40	52.00	59.90	55.90
SAGM [51]	57.70	54.80	51.60	61.40	56.30
ITTA [78]	56.00	51.50	50.30	61.60	54.90
CGS (Ours)	**59.3**	**55.7**	**53.5**	**65.0**	**58.4**

S Single Object Tracking Experiments

We provide additional evidence for the task-adaptability of CGS on single object tracking. We use OSTrack-256 [80] as the base tracker and compare the standard baseline against CGS. Unlike classification, tracking contains a template-search structure, so the destruction probe is adapted to the task: the template is preserved, while the search-region target evidence is disrupted so that gradients computed on the destroyed view emphasize background, distractor, and search-context shortcuts.

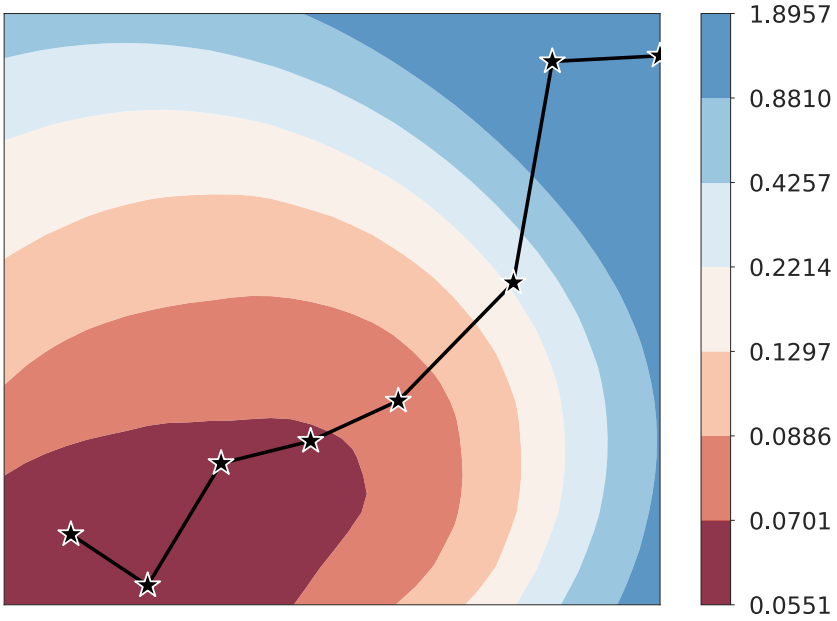


Figure 4: CGS optimization trajectory from the initial weights to the final weights

Table 9: Single object tracking results across GOT-10k, UAV123, and OTB2015. For GOT-10k, we report AO and success rates at thresholds 0.5 and 0.75. For UAV123 and OTB2015, we report AUC.

Method	GOT-10k			UAV123	OTB2015
	AO	$SR_{0.5}$	$SR_{0.75}$	AUC	AUC
MDNet	29.9	30.3	9.9	52.8	67.8
ECO	31.6	30.9	11.1	53.2	69.1
SiamRPN++	51.7	61.6	32.5	61.3	**69.6**
ATOM	55.6	63.4	40.2	64.2	66.9
DiMP	61.1	71.7	49.2	65.3	68.4
PrDiMP	63.4	73.8	54.3	68.0	**69.6**
OSTrack-256 Baseline	71.0	80.4	68.2	65.8	66.68
OSTrack-256 + CGS	**72.4**	81.8	**68.8**	66.14	68.37

The destroyed view is used only to estimate the nuisance-aligned gradient direction; the supervised tracker update remains defined by the clean training pair. The CGS hyperparameters are kept at the same locked configuration used for the plug-in DG and OOD segmentation transfer experiments.

Table 9 summarizes the current tracking results across GOT-10k, UAV123, and OTB2015. On GOT-10k [81], CGS improves AO from 71.0 to 72.4, $SR_{0.5}$ from 80.4 to 81.8, and $SR_{0.75}$ from 68.2 to 68.8. On UAV123 [82] and OTB2015 [83], CGS improves AUC over the OSTrack-256 baseline. These matched base-pipeline results show that the same optimizer-facing principle used in classification can be transferred to tracking when paired with a destroyer that reflects the causal and nuisance structure of the task; we treat them as a transfer demonstration for OSTrack-256 rather than as an exhaustive survey of all trackers.

T Domain-Generalized Semantic Segmentation Experiments

We test CGS on dense prediction by applying the same optimizer-facing principle to SCSD[84], a recent domain-generalized semantic segmentation method. We use shuffle-center-patch destruction only as the optimizer-side probe: the image tensor is destroyed for the auxiliary gradient computation, but the dense label map is not transformed or treated as an augmented supervised target. The

Table 10: Multi-source domain-generalized semantic segmentation on GTAV+SYNTHIA→{Cityscapes, BDD, Mapillary}. We follow the standard SCSD-style matched protocol and report mIoU (%).

Method	Backbone/setting	→C	→B	→M	Avg.
IBN	R-50	35.55	32.18	38.09	35.27
ISW	R-50	37.69	34.09	38.49	36.76
SHADE	R-50	47.43	40.30	47.60	45.11
Pin the Memory	R-50	44.51	38.07	42.70	41.76
AdvStyle	R-50	39.29	39.26	41.14	39.90
TLDR	R-50	48.83	42.58	47.80	46.40
SPC-Net	R-50	46.36	43.18	48.23	45.92
FAMix	R-50	49.41	**45.51**	51.61	48.84
SCSD [84]	CLIP/R-50	52.43	45.25	56.58	51.42
SCSD + CGS	CLIP/R-50	**54.86**	44.52	**57.66**	**52.35**

Table 11: Adverse-condition domain-generalized semantic segmentation on GTAV→ACDC. We report mIoU (%) on night (AN), snow (AS), rain (AR), and fog (AF).

Method	→AN	→AS	→AR	→AF	Avg.
ISW	6.32	29.97	33.02	32.56	25.47
SHADE	8.18	30.38	35.44	36.87	27.72
WildNet	8.27	30.29	36.32	35.39	27.57
SiamDoGe	10.60	30.71	35.84	36.45	28.40
TLDR	13.13	36.02	38.89	40.58	32.16
FAMix	14.96	37.09	38.66	40.25	32.74
SCSD [84]	15.06	41.37	**42.77**	43.43	35.66
SCSD + CGS	**16.45**	**44.37**	42.75	**44.38**	**36.99**

ordinary segmentation loss on the clean image remains the descent objective. We keep the same locked CGS hyperparameter configuration used in the plug-in and tracking experiments; only the task-specific destruction operator changes. Tables 10 and 11 report the two locked segmentation comparisons. Under the same SCSD-style matched protocol, CGS improves the SCSD averages on both GTAV+SYNTHIA→{Cityscapes, BDD, Mapillary} and GTAV→ACDC by +0.93 and +1.33 mIoU, respectively. We use these results to support transfer of the optimizer-side correction to dense prediction, while the main controlled benchmark core remains classification.

T.1 Class-Level Segmentation Results

We report matched class-level IoU changes for representative road-scene categories, including narrow and small foreground objects. The baseline and CGS models use the same GTAV-source training pipeline, architecture, schedule, and hyperparameters, differing only in the CGS correction. Each entry in Table 12 averages the change in IoU over Cityscapes, BDD, and the four ACDC conditions (fog, rain, snow, and night). Object-size bands are based on measured mean ground-truth connected-component areas from Cityscapes and BDD validation masks. The segmentation destroyer shuffles 2×2-pixel tiles within the central image region.

Table 12: Mean class-level ΔIoU (percentage points) across six targets (CGS minus matched GTAV-source baseline). Size bands follow ground-truth connected-component area.

Class	Size band	Mean ΔIoU
Traffic light	Small	+0.79
Traffic sign	Small	+1.23
Bicycle	Small	+10.88
Motorcycle	Small	+2.59
Car	Large	+2.97
Mean	–	**+3.69**

All five listed classes improve in mean IoU, with particularly substantial gains for bicycles. These results show that CGS gains extend to several small-object classes in the matched segmentation setting.

To provide the complete class-level results, Table 13 gives the matched GTAV-source CGS-minus-baseline changes for all 19 semantic classes on each of the six targets. The fixed size bands are determined by mean ground-truth connected-component area, normalized by image area, measured from Cityscapes and BDD validation masks. They are not reselected per target. These within-framework GTAV-source deltas are distinct from the GTAV+SYNTHIA multi-source comparison reported in Table 10.

Table 13: Complete class-level ΔIoU (percentage points, CGS minus our reproduced SCSD baseline) for GTAV-source training. Area (ppm) is mean connected-component area normalized by image area; bands are the smallest seven, middle six, and largest six classes. ACDC columns are fog (F), rain (R), snow (S), and night (N). A dash indicates an unavailable measurement.

Class	Area (ppm)	Band	Cityscapes	BDD	ACDC-F	ACDC-R	ACDC-S	ACDC-N
Traffic light	560	Small	+0.11	-1.25	-0.80	-0.75	-2.17	+9.61
Pole	745	Small	-0.52	+1.54	-2.87	-4.49	-0.11	+0.44
Traffic sign	850	Small	+0.15	+2.48	-0.72	+5.13	+0.84	-0.50
Rider	1,718	Small	-4.02	-0.95	-0.34	-0.05	+1.62	-2.07
Bicycle	1,847	Small	+12.32	+8.52	+32.24	0.00	+9.41	+2.81
Person	2,456	Small	-0.64	-6.29	-4.50	+0.84	+11.35	-1.76
Motorcycle	3,301	Small	+2.66	-4.84	+1.98	+2.62	+12.97	+0.13
Terrain	4,623	Medium	-1.56	-1.28	-1.61	+2.70	+1.03	-2.01
Fence	5,979	Medium	+0.57	+3.65	+2.29	-4.42	+1.99	-5.85
Wall	6,176	Medium	-3.01	+0.06	+5.77	-2.07	+2.73	-3.63
Sidewalk	6,450	Medium	-3.30	+3.85	+3.24	+1.55	-5.55	+8.21
Train	7,313	Medium	-3.02	-0.37	-26.26	-2.90	-10.91	+0.69
Building	12,334	Medium	+1.45	+0.99	+0.80	+0.59	-0.02	+1.78
Car	15,451	Large	-1.55	+4.57	+1.55	+2.94	+14.52	-4.22
Truck	15,571	Large	-4.58	-2.15	-1.09	+4.15	+11.72	-1.17
Vegetation	18,912	Large	+0.12	+0.35	-1.83	-0.32	-2.02	-13.38
Bus	19,394	Large	-3.32	+6.91	+19.12	+2.57	-5.56	–
Sky	24,026	Large	+5.46	-0.16	-1.71	-0.82	-2.80	-0.02
Road	61,856	Large	-1.13	+4.05	+3.00	+2.39	-1.63	+5.57
Small mean	–	–	**+1.44**	**-0.11**	**+3.57**	**+0.47**	**+4.84**	**+1.24**
Medium mean	–	–	**-1.48**	**+1.15**	**-2.63**	**-0.76**	**-1.79**	**-0.13**
Large mean	–	–	**-0.83**	**+2.26**	**+3.17**	**+1.82**	**+2.37**	**-2.64**
mIoU Δ	–	–	**-0.20**	**+1.03**	**+1.49**	**+0.50**	**+1.97**	**-0.30**

The small-object band has a positive mean change on five of six targets, with a slight decrease on BDD. Individual class changes vary, so these descriptive size-band comparisons do not establish uniform improvement for every small object.

U Ablations & Sensitivity

We ablate the four algorithmic hyperparameters we directly control: intervention frequency (k), grad-inhibition strength (γ), patch size, and the shuffle mechanism. To save space, we report per-domain accuracies using the abbreviations: OfficeHome (**OH**), TerraIncognita (**TI**), DomainNet (**DN**), PACS (**PA**), and VLCS (**VL**). We also report the macro average and the improvement relative to our canonical configuration in a single column formatted as **Avg (Δ)**. Unless noted, each ablation varies a single hyperparameter while the others are fixed to one canonical hyperparameter tuple used across all datasets (*Size=2, k=100, Mechanism=shuffle center patches,* $\gamma = 0.5$). This shared-tuple setting gives the 68.0 canonical ablation average in Table 14. The 69.0 main-table average follows the SWAD-style protocol [48]: select a single hyperparameter value per dataset rather than per target domain, which is still stricter than selecting hyperparameters separately for each held-out domain.

Ablations Experimental Setup Tables 15, 16, 17, and 18 each vary a single hyperparameter. For each knob we sort by DG-Avg, and report Δ DG-Avg relative to the canonical configuration. The below paragraphs discuss the effect of changing each hyperparameter individually. Afterwards we discuss the cross-domain consistency of our ablations, then we discuss the overall sensitivity of the algorithm to hyperparameter changes.

Mechanism-facing controls. The ablations below are hyperparameter sweeps, but they also separate CGS from ordinary augmentation in two ways. First, the destroyed branch never contributes a supervised descent loss; it only supplies a gated direction for modifying the clean gradient. Second, the gate requires clean/destroyed gradient agreement, so a probe that behaves like unstructured noise

Table 14: Canonical configuration results (Size=2, k=100, Mechanism=center, $\gamma = 0.5$).

Config	OH	TI	DN	PA	VL	Avg
Baseline	73.4	51.4	49.8	87.8	77.7	68.0

Table 15: Ablation on intervention frequency (k). We report per-domain results and the Avg (Δ) relative to the canonical configuration.

k	OH	TI	DN	PA	VL	Avg (Δ)
800	73.5	52.1	49.9	87.9	78.4	68.4 (+0.3)
200	73.3	51.5	49.7	88.5	78.2	68.2 (+0.2)
600	73.2	52.0	50.1	87.6	78.1	68.2 (+0.2)
400	73.5	51.4	49.4	88.0	78.0	68.1 (+0.0)

is suppressed rather than optimized. The shuffle-mechanism sweep tests whether the effect depends on one hand-picked corruption, and the qualitative sections that follow visualize whether the probes destroy global object evidence while leaving local nuisance statistics visible.

Intervention Frequency (k) Varying only *intervention frequency (k)* yields a best DG-Avg of 68.4 at intervention frequency (k)=800 (Δ +0.3 pp vs. baseline). A robust band spans [200, 400, 600, 800] ($\leq$1.0 pp from best). Domain sensitivity ranges (max-min across the sweep) indicate PACS is most sensitive (0.9 pp), while OfficeHome is relatively stable (0.3 pp).

Grad-Inhib Strength (γ) Varying only *grad-inhib strength (γ)* yields a best DG-Avg of 68.2 at grad-inhib strength (γ)=1.2 (Δ +0.1 pp vs. baseline). A robust band spans [0.8, 1.0, 1.2, 1.5] ($\leq$1.0 pp from best). Domain sensitivity ranges (max-min across the sweep) indicate PACS is most sensitive (0.4 pp), while OfficeHome is relatively stable (0.1 pp). The theoretically derived proximal update corresponds to $\gamma \in [0, 1)$ and the default configuration uses $\gamma = 0.5$; values $\gamma > 1$ are reported only as over-relaxed engineering variants and are not used by the proximal proof.

Patch Size Varying only *patch size* yields a best DG-Avg of 68.3 at patch size=4 (Δ +0.3 pp vs. baseline). A robust band spans [4, 6, 8, 10] ($\leq$1.0 pp from best). Domain sensitivity ranges (max-min across the sweep) indicate DomainNet is most sensitive (1.5 pp), while OfficeHome is relatively stable (0.2 pp).

Shuffle Mechanism Varying only *shuffle mechanism* yields a best DG-Avg of 68.1 at shuffle mechanism=Phase (Δ +0.1 pp vs. baseline). A robust band spans ['Phase', 'Voronoi'] ($\leq$1.0 pp from best). Domain sensitivity ranges (max-min across the sweep) indicate VLCS is most sensitive (5.1 pp), while TerraIncognita is relatively stable (0.0 pp).

Cross-domain consistency We correlate per-run deltas with respect to the canonical configuration and observe that datasets with higher appearance variability (TerraIncognita, DomainNet) tend to co-move, whereas PACS remains comparatively stable. This pattern explains why frequency and γ exhibit wide robust bands: they regularize spurious cues without suppressing class-discriminative gradients, benefiting diverse datasets while keeping PACS steady. In practice we find that the harder datasets (TerraIncognita, DomainNet) tend to move together, while PACS is more stable, mirroring qualitative trends in the sensitivity curves.

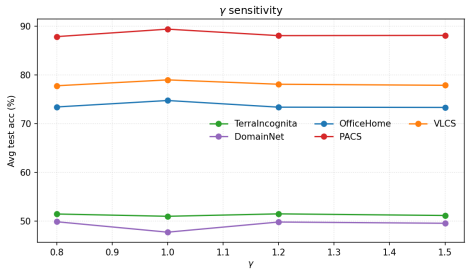
Sensitivity figures We include the sensitivity plots (k, γ, Size) in figure 5.

Table 16: Ablation on grad-inhibition strength (γ). Other hyperparameters are fixed to the canonical configuration. Values above one are over-relaxed variants, not proximal-penalty settings.

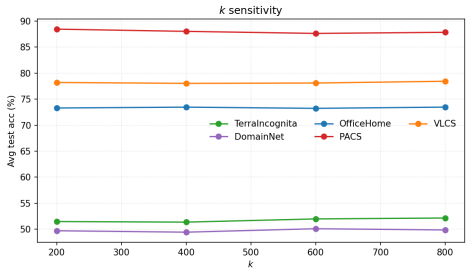
γ	OH	TI	DN	PA	VL	Avg (Δ)
1.2	73.4	51.5	49.8	88.0	78.1	68.2 (+0.1)
1.0	73.4	51.3	49.9	88.2	77.9	68.1 (+0.1)
0.8	73.4	51.4	49.8	87.8	77.7	68.0 (+0.0)
1.5	73.3	51.1	49.5	88.1	77.8	68.0 (-0.1)

Table 17: Ablation on patch size. Other hyperparameters fixed to canonical.

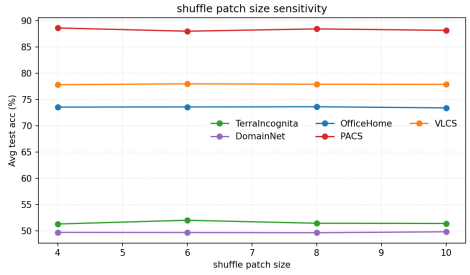
Size	OH	TI	DN	PA	VL	Avg (Δ)
4	73.6	51.3	49.7	89.1	77.8	68.3 (+0.3)
10	73.4	51.4	49.8	88.7	77.9	68.2 (+0.2)
8	73.6	51.4	49.6	88.4	77.9	68.2 (+0.2)
6	73.6	51.7	48.3	88.0	78.0	67.9 (-0.1)



(a) γ



(b) k



(c) Patch size

Figure 5: Sensitivity analyses for CGS hyperparameters. Each subfigure shows how a single hyperparameter affects DG performance.

U.1 Component Removal and Stress Tests

The preceding sweeps show how CGS responds to different hyperparameter values. Here we change individual parts of the steering rule itself, using six matched tests on TerraIncognita and PACS. The full-method reference is 52.70 and 89.10, respectively; the earlier shared-tuple sweeps use the separate canonical configuration in Table 14.

Table 18: Ablation on shuffle mechanism at matched settings.

Mech.	OH	TI	DN	PA	VL	Avg (Δ)
Voronoi	73.8	50.7	49.8	88.1	73.8	67.2 (-0.8)
Phase	73.7	50.7	49.2	88.1	78.9	68.1 (+0.1)

Table 19: Component removal and stress tests: classification accuracy (%) and change (pp) relative to full CGS on TerraIncognita (TI) and PACS.

Configuration	TI	ΔTI	PACS	ΔPACS
Full CGS	**52.70**	–	**89.10**	–
Probe every step ($k = 1$)	48.14	-4.56	85.85	-3.25
Zero gate threshold ($\tau = 0$)	49.62	-3.08	87.39	-1.71
No damping ($\gamma = 1$)	49.63	-3.07	87.06	-2.04
Steering disabled ($\gamma = 0$)	49.86	-2.84	87.16	-1.94
No layerwise renormalization	50.14	-2.56	87.16	-1.94
Global projection	51.33	-1.37	86.96	-2.14

The largest decline occurs when the destroyed-view probe runs at every step ($k = 1$): 4.56 points on TerraIncognita and 3.25 on PACS. The reported moderate steering strength ($\gamma = 0.5$) also outperforms both disabling steering and setting $\gamma = 1$. The gate and layerwise operations each contribute to the full configuration, supporting a correction that is selective in both timing and direction.

U.2 Held-Out Destroyer Audit and Practical Selection

We choose a destroyer to change what supports the model's prediction, not merely to make the image look corrupted. Patch permutation breaks shape and part layout in classification while keeping local color and texture. Finer 2×2-pixel tiles disrupt spatial continuity in segmentation; tracking instead preserves the template while disrupting target evidence in the search region. Each task therefore uses the same principle at a different scale.

Held-out linear probes. Using the frozen ImageNet-pretrained backbone used to initialize CGS, we fit separate linear probes for class label y and source-domain label d on clean representations $\phi(x)$ and destroyed representations $\phi(\mathcal{P}(x))$. The probes for destroyed views are trained on destroyed features themselves, rather than evaluated out-of-distribution after training on clean features. We report held-out accuracy for the submitted patch-shuffling operator:

Table 20: Held-out class- and domain-probe accuracy (%) before and after the submitted patch-shuffling destroyer.

Dataset	Representation	Class y	Domain d
OfficeHome	Clean	76.7	70.4
	Destroyed	47.4	67.2
PACS	Clean	85.3	97.4
	Destroyed	68.3	94.2

After shuffling, the OfficeHome class probe loses 29.3 points of accuracy, compared with 3.2 points for the domain probe. PACS shows the same separation (17.0 versus 3.2 points). The destroyed views retain substantial domain-related information even as class prediction becomes markedly harder.

U.3 Controlled Spurious-Correlation Benchmarks

We evaluate CGS on ColoredMNIST, Waterbirds, and CelebA against matched ERM baselines using each benchmark's evaluation protocol. We report worst-group accuracy, the lowest classification accuracy across the benchmark's predefined label–shortcut groups.

Table 21: Worst-group accuracy (%) on controlled spurious-correlation benchmarks with matched ERM comparisons.

Benchmark	ERM	CGS	Gain
ColoredMNIST	51.5	**63.8**	+12.3
Waterbirds	60.0	**87.0**	+27.0
CelebA	41.1	**47.0**	+5.9

The largest improvement is on Waterbirds, where backgrounds can serve as strong cues for bird class. The gains on ColoredMNIST and CelebA show that the effect extends beyond this foreground–background setting: on CelebA, the target and shortcut are both facial attributes.

U.4 Object-Preserving Background Intervention

To measure sensitivity to scene context while keeping the foreground object fixed, we placed 130 OfficeHome Product objects spanning all 65 classes onto natural-scene backgrounds. A neutral-grey background provides a control for image compositing. We report classification accuracy on the original images, grey-background composites, and natural-scene composites, along with the scene-induced accuracy drop and prediction flip rate.

Table 22: OfficeHome Product background intervention: accuracy and prediction flip rate (%). The scene drop is measured in percentage points relative to original-image accuracy.

Model	Original	Grey	Scene	Scene drop	Flip rate
ERM	96.9	90.0	75.6	21.3	22.0
CGS	96.9	**90.8**	**79.0**	**17.9**	**18.5**

The models have identical original-image accuracy, but replacing the background with a natural scene produces a 3.4-point smaller accuracy drop for CGS and a 3.5-point lower prediction-flip rate. Their grey-background accuracies are close (90.0 and 90.8), while the natural-scene composites separate the methods more clearly (75.6 and 79.0).

V Qualitative Analysis: Visualization of Intervention Probes

Figure 6 illustrates the concrete probe families used in our analysis and ablations. The key point is not that these images are realistic counterfactuals or label-preserving training examples. They are diagnostic views designed to suppress enough semantic evidence to expose shortcut-sensitive gradients while preserving nuisance statistics such as texture, color, local layout, or background fragments.

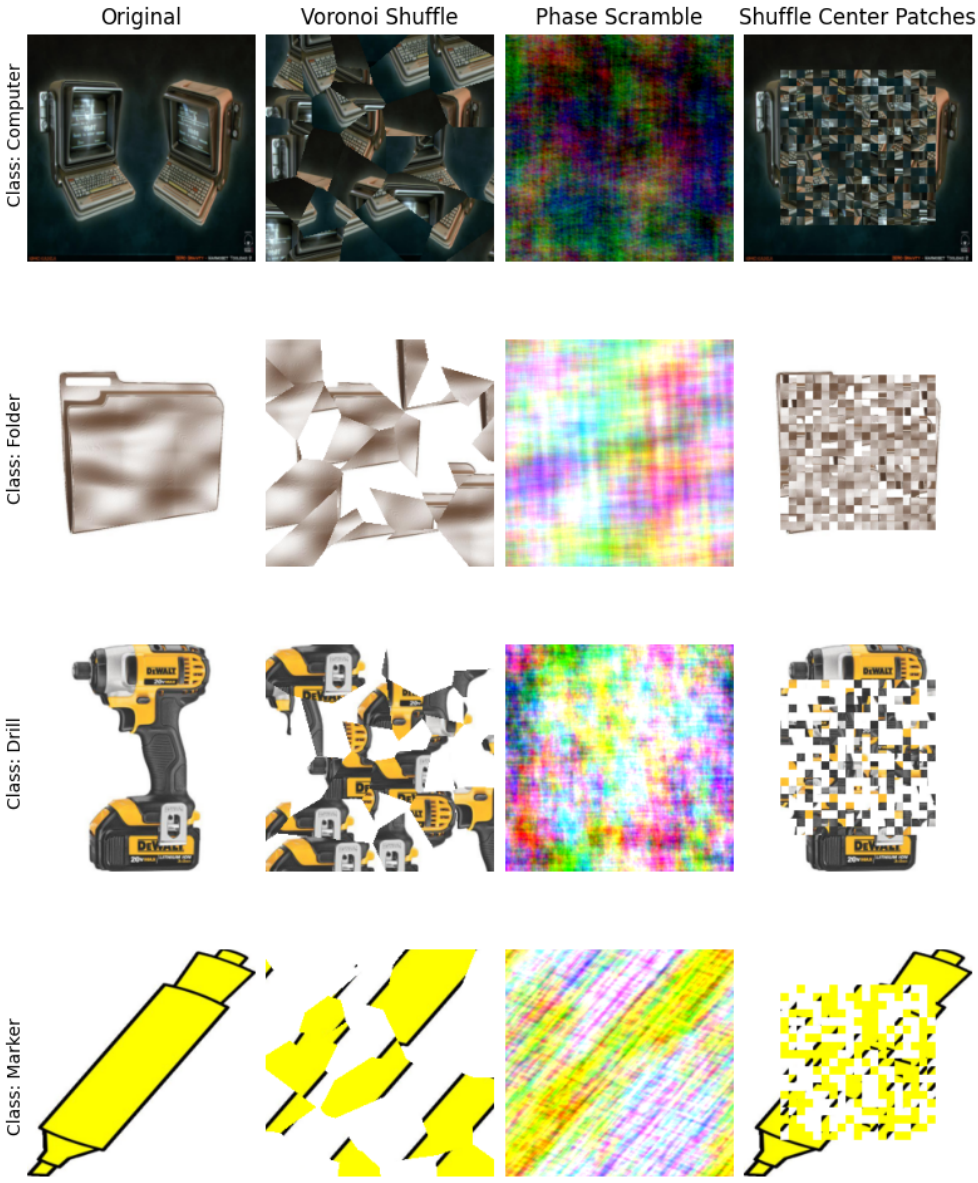


Figure 6: Visual examples of different intervention probes applied to a sample image. The figure shows the original image together with representative probe families including shuffle-center-patches, phase perturbation, and Voronoi-style shuffling. These probe-transformed images are used not for training, but as analytical probes to estimate a nuisance-sensitive cone normal.

W Qualitative Analysis: Loss Landscape Visualization

To provide empirical support for the geometric intuition that CGS encourages convergence to flatter minima (as discussed in Proposition C.2), Figure 7 visualizes 2D projections of the loss surface around the CGS-trained solution on OfficeHome, where we train on 3 domains and test on one held-out unseen domain. A side-by-side ERM landscape would be a useful additional diagnostic, but is not included in the current figure.

The visualization offers a clear insight. The landscape corresponding to the CGS-trained model consistently exhibits a wide, flat basin of low loss across the different unseen target domains. This flatness indicates that the model's performance is less sensitive to small perturbations in its parameters, a characteristic that is strongly correlated with improved generalization. The visualization is therefore qualitative support for the claim that the CGS gradient steering mechanism can guide the optimization trajectory toward solutions that are less sensitive to small parameter perturbations.

Beyond the static endpoint landscape in Figure 7, it is also informative to visualize how CGS reaches that flat region during training. Figure 4 plots the empirical risk over a 2D slice of parameter space, with the CGS optimization path overlaid as a sequence of star markers from the initial weights to the final weights. The trajectory descends monotonically through the contour levels and terminates inside the wide low-loss basin highlighted earlier in this section. Two qualitative features are worth noting. First, the path does not overshoot into the steep outer region of the contour map, consistent with the cone-projection step suppressing nuisance-aligned descent directions that would otherwise pull the iterate toward sharper minima. Second, the late-stage iterates concentrate inside the lowest contour band rather than oscillating along its rim, which is the behavior expected from the curvature-suppression argument in Proposition C.2: once the cone-normal direction is removed, the remaining tangent subspace has strictly lower worst-case curvature whenever the cone normal overlaps the sharpest eigendirection of the Hessian.

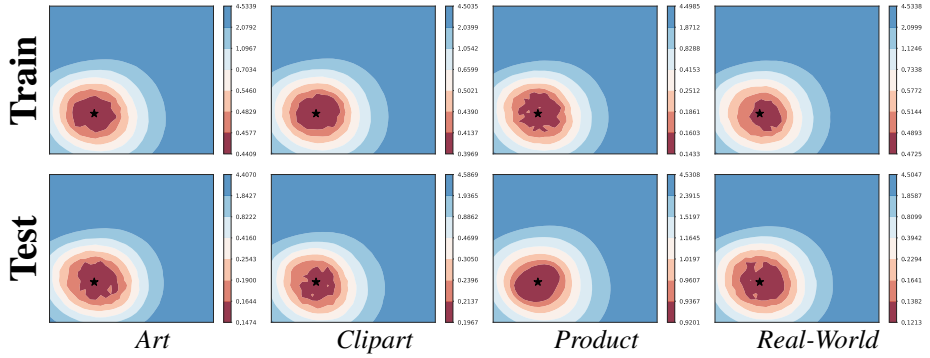


Figure 7: Loss surfaces on model parameters in the Office-Home dataset for each target domain. The center star mark indicates the optimal model parameters, and the losses on the plane are visualized with contours. The first and second rows show the averaged training loss and the test loss surfaces, respectively. This figure illustrates that the CGS-trained weights lie in a flat region in the loss landscape, leading to better generalization.

X Qualitative Analysis: Visualizing Model Attention

To gain a deeper, intuitive understanding of why CGS improves generalization, we investigate what features the learned models use for prediction. We employ Grad-CAM [85] to generate visual explanations for the predictions of both a standard ERM model and a CGS-trained model. This lets us compare whether the models focus primarily on object-relevant evidence or rely more heavily on nuisance background cues.

As illustrated in Figure 8, the ERM model frequently makes incorrect predictions by focusing on irrelevant background textures or surrounding context. For instance, in the second column it misclassifies a Drill as a Trash Can because its attention is captured by the metallic textured surface behind the object. Similarly, for the Alarm Clock and Oven examples, its attention spreads to surrounding context rather than concentrating on the object itself.

In contrast, the CGS-trained model consistently places more of its attention on the object of interest and less on distracting background structure. This targeted focus coincides with better classifications in these examples. These visualizations therefore provide qualitative support for the nuisance-steering interpretation of CGS: the cone-aware correction appears to reduce reliance on nuisance background cues and shift the model toward more object-centered evidence.

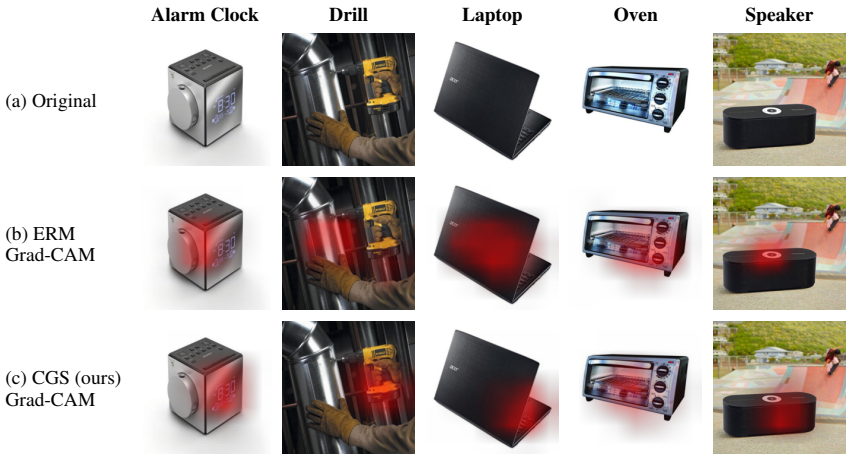


Figure 8: Qualitative comparison of Grad-CAM visualizations across objects (columns) for each visualization type (rows): (a) Original, (b) ERM, and (c) CGS (ours). ERM often places substantial attention on nuisance background structure and misclassifies, while CGS concentrates more on the object region and classifies correctly in these examples.

Y Responsible Research Considerations

Limitations. CGS is designed for nuisance-dominant distribution shifts in which a destruction probe can suppress semantic evidence while preserving nuisance structure. Its effectiveness may decrease when the probe leaks too much semantic information, destroys the nuisance cues that should be measured, or when the target shift is dominated by semantic selection bias rather than nuisance variation. The controlled benchmark core of this paper remains image classification with a ResNet-50 backbone, while the tracking and segmentation experiments demonstrate transfer under a shared CGS hyperparameter configuration rather than a full foundation-model or modality-wide study. Extending the theory and evaluation to broader architectures, modalities, and stronger semantic selection bias remains future work.

Reproducibility and statistical reporting. The experiments use public DG benchmarks through DomainBed, an ImageNet-pretrained ResNet-50 backbone, and the fixed hyperparameters reported in Section 3 and Appendix O. Appendix O also gives the classification command schema, locked CGS hyperparameter JSON, seeds, software versions, and hardware used for the main reproducibility path. When repeated runs are available, tables report mean accuracy with standard deviation; baseline uncertainty values are taken from the cited papers or established benchmark reports. A reported ± 0.0 means the standard deviation is below 0.05 after rounding, not that the run is variance-free. The method is fully specified by Algorithms 1 and 2: clean gradients are computed as usual, destroyed gradients are computed every k steps, the raw dot gate selects aligned layers, and the layerwise gradient is projected and norm-rescaled before the optimizer step.

Compute. CGS has near-ERM per-step cost: it uses the ordinary forward/backward pass and adds one extra forward/backward pair every k steps to estimate the nuisance direction. Table 6 compares this overhead with other gradient-based DG methods; with the default $k = 100$, the average pass-count overhead is $1.01\times$ ERM. Appendix Q additionally reports a measured ERM-vs-CGS wall-clock comparison, including step time and peak GPU memory. Because CGS changes only the training update, inference uses the unmodified base network and has the same test-time cost as ERM or the wrapped method.

Assets and licenses. The work uses existing public datasets and software assets, including PACS, VLCS, OfficeHome, TerraIncognita, DomainNet, DomainBed, ImageNet-pretrained ResNet-50 weights, and standard PyTorch-style training components. The corresponding papers and packages are cited throughout the manuscript, and the method does not introduce a new dataset or redistribute third-party benchmark assets.

Broader impacts and safeguards. Improving out-of-distribution generalization can make vision systems more reliable under distribution shift, which is beneficial for scientific and applied settings. However, benchmark robustness does not guarantee safe deployment in sensitive domains such as medical, surveillance, biometric, or safety-critical applications; practitioners should validate performance under the intended deployment distribution, audit subgroup behavior, and avoid using the method as a substitute for application-specific safety analysis. The paper does not release a high-risk pretrained model, generative model, scraped dataset, or human-subject data, so no special release safeguards are required beyond responsible use of the public benchmarks and code.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and Section 1 state the paper's optimizer-facing claims, assumptions, and contributions, while Sections 2–3 and the appendix provide the corresponding theory and empirical evidence.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: The conclusion states the current scope of the theory and experiments, and Appendix Y discusses the main limitations: nuisance-dominant classification shifts, dependence on the destruction probe, and the need to extend to stronger semantic selection bias and dense prediction.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [Yes]

Justification: The theoretical results state their assumptions in Section 2 and Appendix B; complete proof details are provided in Appendices C, E, G, J, and L.

Guidelines:

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

4. **Experimental result reproducibility**

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

Answer: [Yes]

Justification: Section 3 identifies the evaluation datasets and summarizes the DomainBed-style protocol. Appendix O documents the leave-one-domain-out and training-domain validation procedures, backbone, optimizer, batch size, learning rate, and locked CGS hyperparameters, along with the reproduction command, seeds, software versions, hardware, and per-domain results.

Guidelines:

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

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [Yes]

   Justification: An anonymized code package is provided as supplementary material. The paper uses public benchmarks and provides detailed algorithms, hyperparameters, command schema, environment details, and hardware information for reproducibility.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: Section 3 identifies the benchmark datasets and summarizes the evaluation protocol. Appendix O specifies the DomainBed training and validation protocol, backbone, optimizer, batch size, learning rate, and locked CGS hyperparameters; Appendix U reports hyperparameter sensitivity.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

Justification: For the main benchmark table, uncertainty values for prior methods are source-reported when available, while CGS uncertainty is computed as the standard deviation over 3 repeated runs under the official DomainBed protocol.

Guidelines:

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Appendix O reports the GPU, CPU, RAM, software environment, and main classification run settings; Appendix Q and Table 6 report the theoretical pass-count overhead, peak GPU memory, and a measured ERM-vs-CGS wall-clock comparison.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The work uses public image-classification benchmarks and does not involve human subjects, private data, or high-risk model releases; Appendix Y summarizes responsible-use considerations and remaining risks.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: Appendix Y discusses positive impacts from improved robustness and negative risks from over-reliance or deployment in sensitive settings without task-specific validation and fairness auditing.

    Guidelines:

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

11. **Safeguards**

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

    Answer: [N/A]

    Justification: The paper does not release a high-risk pretrained model, generative model, scraped dataset, or other asset requiring special misuse safeguards; it introduces an optimization method evaluated on existing public benchmarks.

    Guidelines:

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

12. **Licenses for existing assets**

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

Answer: [Yes]

Justification: The paper cites the original sources of the existing datasets, software, and pretrained models used in the experiments. We respect their applicable terms of use.

Guidelines:

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

13. **New assets**

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

Answer: [N/A]

Justification: The paper introduces a method and theoretical analysis, but it does not introduce or release a new dataset, benchmark, model checkpoint, or other standalone asset requiring separate asset documentation.

Guidelines:

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

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

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

Answer: [N/A]

Justification: The experiments use existing public image-classification benchmarks and do not involve crowdsourcing or research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

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

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

    Answer: [N/A]

    Justification: The work does not involve crowdsourcing or human-subject research, so IRB approval or equivalent human-subjects review is not applicable.

    Guidelines:

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

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: The core method, theory, and experiments do not use LLMs as an important, original, or non-standard component of the research.

    Guidelines:

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