You Can't Have It Both Ways: Concept Entanglement Limits Diffusion Model Unlearning

Yian Wang[1], **Ali Ebrahimpour-Boroojeny**[1,2,*], **Hari Sundaram**[1], **Varun Chandrasekaran**[1]

[1]University of Illinois Urbana-Champaign
[2]Google

Abstract

Concept unlearning in text-to-image diffusion models aims to suppress a target concept (e.g., `horse`) while preserving related but distinct content (e.g., `donkey`), yet existing methods either leak under indirect prompts or visibly degrade other concepts. We show that these failure modes stem from the geometry of concept representations rather than from any particular algorithm. Formalizing concepts as activation-space regions, we prove that the overlap between a target and other concepts lower-bounds the damage any robust erasure must inflict on them, with the trade-off scaling linearly in the degree of overlap. Across thirteen unlearning methods, including methods designed to preserve non-target concepts, no method achieves both strong erasure and strong neighbor preservation: STEREO nearly eliminates indirect leakage but cuts neighbor generation by more than 75%, while sparse inference-time methods preserve neighbors but leak. Damage increases with our overlap measure, monotonically so for STEREO; the κ-scaling reproduces on SDXL, and neighbor-selective damage recurs on FLUX. Perfect unlearning is the wrong target for entangled concepts; methods should be evaluated on the Pareto frontier our theorem establishes.

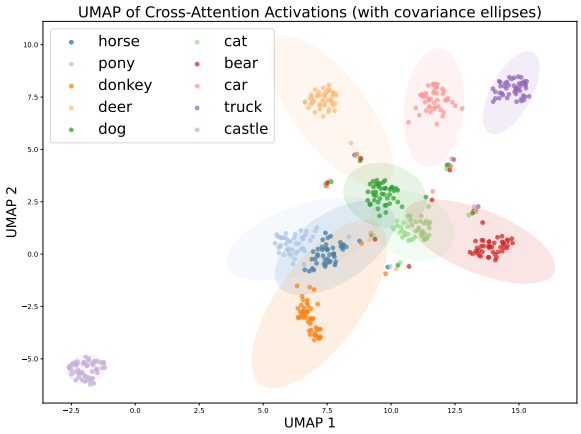
1 Introduction

Text-to-image diffusion models [Ho et al., 2020, Rombach et al., 2022, Saharia et al., 2022] have rapidly become the dominant generative paradigm for high-fidelity image synthesis, with deployments ranging from creative tools to commercial APIs. Privacy concerns and copyrights have motivated active research on *concept unlearning* [Gandikota et al., 2023]: given a trained model and an undesirable concept (e.g., an object category, an artistic style, or a specific identity), the goal is to suppress the concept while preserving the model's general capabilities. In this work, we focus on object-category concepts, where benchmarks and detectors are most mature. A growing body of work has proposed methods for this task, ranging from fine-tuning-based approaches that permanently alter model weights [Gandikota et al., 2023, Fan et al., 2023, Zhang et al., 2024, Srivatsan et al., 2025] to inference-time interventions that block concept-specific activations without modifying the underlying model [Li et al., 2024, Lyu et al., 2024, Cywiński and Deja, 2025]. These methods are typically evaluated on two families of metrics: *unlearning accuracy* (UA), measuring whether the target concept has been successfully suppressed, and *retain accuracy* (RA), measuring whether the model's performance on non-target concepts remains intact.

A tacit assumption in this evaluation paradigm is that UA and RA can be optimized independently. However, our empirical evidence suggests this assumption is violated (Section 3). Methods achieving strong erasure (high UA), such as STEREO [Srivatsan et al., 2025], often degrade generation quality on semantically related concepts (low RA). Methods with better retention, such as SAeU-

[*]This work was conducted while at the University of Illinois Urbana-Champaign.

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



Concept	$\hat{\kappa}$	Neighbors
dog	0.89	puppy, labrador
bear	0.82	grizzly, panda
horse	0.81	donkey, foal, mare
cat	0.77	kitten, lynx
castle	0.27	fortress, palace, citadel

Figure 1: **Concepts occupy coherent, partially overlapping regions in activation space.** *Top-left:* UMAP projection of cross-attention activations at `mid_block.attentions.0` (step 25/50) for 10 concepts in Stable Diffusion v1.4 (visualization pool; full evaluation pool in Appendix I.3); each point is one prompt and ellipses show covariance contours. Semantically related concepts (`horse`/`pony`/`donkey`) cluster together, while isolated concepts (`castle`) are separated, at the bottom left of the figure. *Bottom-left:* Estimated entanglement coefficient $\hat{\kappa}$ (fraction of the target's activation region shared with semantic neighbors; see Definition 4.1). The Neighbors column lists representative discovered neighbors. Animal subtypes have high $\hat{\kappa}$ and dense neighbor sets; `castle` has low $\hat{\kappa}$ despite a comparable number of neighbors. Ellipses are $(\mathbf{z}-\boldsymbol{\mu}_c)^\top \boldsymbol{\Sigma}_c^{-1}(\mathbf{z}-\boldsymbol{\mu}_c) \leq 4$ for the mean and covariance of each concept's projected points; they are visualization aids only and are not used to compute $\hat{\kappa}$.

ron [Cywiński and Deja, 2025], leave the target concept reachable through prompts that omit its name but supply correlated context (e.g., generating a horse from "a jockey at the racetrack"), a phenomenon we call *concept leakage*, resulting in low UA. These observations indicate a fundamental tension: *unlearning and retention are inherently coupled when concepts share representational structure*, a coupling visible directly in the model's internal representations, where semantically related concepts (`horse`/`pony`/`donkey`) occupy overlapping regions of activation space while isolated concepts (`castle`) sit well-separated (Figure 1). This shows that concept activation regions are real geometric structures whose overlap reflects semantic similarity, motivating our formal notion of concept activation regions in Section 4[2]. Two failure modes follow from this geometry: *concept leakage*, where the target concept reappears under indirect prompts, and *collateral forgetting*, degraded generation of semantically related but distinct concepts. We show that these are not shortcomings of individual algorithms but dual symptoms of overlapping concept regions, and that any robust erasure method must trade them off.

Contributions. **(i)** We give a geometric account of concept entanglement, showing that the literature's two failure modes, concept leakage and collateral forgetting, are dual symptoms of overlapping concept regions in cross-attention activation space (quantified by an entanglement coefficient $\hat{\kappa}$), rather than separate algorithmic shortcomings as treated in prior work [Gandikota et al., 2023, Zhang et al., 2024, Srivatsan et al., 2025, Cywiński and Deja, 2025]. **(ii)** Building on this account, we derive a geometric lower-bound for concept unlearning under empirically-verified assumptions: any method achieving ε-robust erasure must incur utility preservation error $\gamma \geq \kappa(1-\varepsilon)/L$ (Theorem 4.6), with strict impossibility at $\varepsilon = 0$ whenever $\kappa > 0$ (Corollary 4.8); this result reframes unlearning as a Pareto problem rather than a dual-objective one. **(iii)** We validate the predicted κ-scaled trade-off across thirteen methods on Stable Diffusion v1.4 (all methods on five targets, four methods on five additional targets that fill the intermediate $\hat{\kappa}$ range), with κ-scaling reproduced on SDXL, using a new Neighbor Preservation metric tied to the activation-level γ in our theorem.

[2]The formal definition of concept activation regions is in Appendix A; full empirical construction details and per-concept clustering statistics are in Appendix B.

2 Related Work

Machine unlearning in diffusion models aims to remove a target concept c (e.g., *horse*) from a pretrained model $\mathcal{D}_\theta$ while preserving overall generative capabilities. The goal is an updated model $\mathcal{D}_{\hat{\theta}}$ that suppresses c on target-associated prompts while maintaining quality on unrelated prompts, following prior work on concept erasure in diffusion models. Background on latent diffusion models is in Appendix A.1.

Closed-form editing. UCE [Gandikota et al., 2024] and RECE [Gong et al., 2024] modify cross-attention projections through analytic updates, without gradient training. RECE additionally searches for embeddings that regenerate the target and removes them in closed form.

Fine-tuning methods. A large body of work erase concepts by directly modifying the weights of a pretrained diffusion model. Erased Stable Diffusion (ESD) Gandikota et al. [2023] uses negative guidance to align target-token prompts with a neutral anchor. Subsequent work improves the precision by editing cross-attention representations [Wang et al., 2025] or updating salient concept-related parameters [Fan et al., 2023, Wu and Harandi, 2024]. CPE [Lee et al., 2025] trains residual attention gates with an anchoring loss, and TRUST [Kori et al., 2026] selectively fine-tunes target-associated neurons under Hessian-based regularization. More recent robust methods such as AdvUnlearn Zhang et al. [2024] and STEREO Srivatsan et al. [2025] further defend against prompt-based regeneration attacks. However, these methods still largely define erasure through target tokens, prompts, or localized parameter subsets. They therefore suppress direct target prompts well, but struggle with *concept leakage* through correlated context; when made more aggressive, their edits can propagate through shared representations and cause *collateral damage* to semantic neighbors.

Inference-time methods. A complementary line of work avoids modifying $\mathcal{D}_\theta$ and instead intervenes during generation by manipulating text embeddings [Li et al., 2024], projecting prompt tokens away from a target subspace (SAFREE, Yoon et al., 2025), adding safety guidance to the denoising process (SLD, Schramowski et al., 2023), inserting lightweight concept-blocking adapters [Lyu et al., 2024], or ablating sparse-autoencoder features [Cywiński and Deja, 2025]. These methods are computationally efficient and often preserve broad model behavior. Intervening at inference time does not by itself make a method conservative: depending on how broadly the intervention displaces activations, these methods span the full range from strong erasure with substantial neighbor damage to strong preservation with residual leakage (Section 5.2).

Preservation-oriented methods. Several methods explicitly target the collateral damage we study. CPE anchors non-target concepts during erasure, TRUST restricts updates to target-associated neurons, and SAFREE adapts its filtering to preserve prompt fidelity. Adversarial preservation [Bui et al., 2024] identifies and protects the concepts most sensitive to erasure, Receler [Huang et al., 2024] and MACE [Lu et al., 2024] localize edits through lightweight erasers and LoRA fusion, and meta-unlearning [Gao et al., 2025] prevents relearning of erased concepts. We evaluate CPE, TRUST, and SAFREE; they reach more favorable operating points but do not escape the trade-off (Section 5.2).

Positioning. That aggressive erasure degrades utility is well documented, and the preservation-oriented methods above aim to reduce it; localization work such as Zarei et al. [2026] identifies where concepts are represented. Neither models target-neighbor overlap or derives limits on unlearning. Our claim is narrower and structural: robust erasure necessarily degrades related concepts *because* their representations overlap, a limit we quantify through κ, prove unavoidable (Theorem 4.6, Corollary 4.8), and turn into a frontier-based evaluation protocol. Across all three families, methods implicitly treat concepts as localized and independently suppressible; this assumption fails whenever activation regions overlap.

3 Motivation

Current concept unlearning methods are designed to suppress a concept token (e.g., "`horse`") so as to remove the corresponding visual concept without substantially affecting other concepts. This design implicitly assumes that visual concepts are represented as localized, token-aligned features in the model's joint text-image space. In this section, we present empirical evidence that this assumption

fails: in practice, unlearning a target concept produces *collateral damage* to the concepts with which it is entangled. [3] The severity of this damage is coupled to both the strength of the erasure applied to the target concept and the extent of its entanglement with other concepts. This coupling is the phenomenon our theoretical framework analyzes in Section 4.

The training mechanisms of text-to-image diffusion models make entanglement inevitable. First, diffusion models condition on prompt embeddings produced by text encoders that often exhibit bag-of-words behavior [Yuksekgonul et al., 2022]. As a result, prompts that omit a concept token (e.g., `horse`) but retain its characteristic context (e.g., `jockey`, `saddle`, `racetrack`) can still induce activations associated with that concept (see Figure 7). This implies that the visual concept is not localized in a single token, but is instead distributed across multiple correlated context tokens. Second, semantically related concepts share much of this contextual structure (e.g., *a horse in a field"* vs. *a donkey in a field"*), which leads to overlapping activation regions in the model's internal representation space. Whether this overlap is already present in text space depends on the encoder. Under our estimator, SDXL's CLIP text embeddings place nine of ten concepts in overlapping regions, whereas the T5 encoder of FLUX shows no measurable overlap for any of our five targets (Appendix L). Entanglement in the denoiser therefore need not be inherited from the text encoder; its source and severity depend on both the text representation and the denoiser computation. We treat this comparison as descriptive rather than as a causal decomposition.

Figure 2 illustrates an example of the trade-off due to this entanglement. It shows two unlearning methods using `horse` as the erasure target. Method that perform localized erasure (ESD Gandikota et al. [2023]) successfully suppress generation when the target token is named, but regenerate the concept through indirect prompts that name only context words, which we call *concept leakage*. Methods that erase more aggressively (STEREO Srivatsan et al. [2025]) suppress this leakage but produce visibly degraded outputs for some semantically similar concepts, such as `donkey` and `zebra`, that are entangled with the target concept (see Figure 1). We call this effect *collateral damage*.

Table 1 quantifies this coupling. Averaged over five targets, STEREO achieves near-complete erasure (UA $= 98.5$, IRR $= 3.5$) but retains only NP $= 18.0$ on semantic neighbors, whereas SEOT preserves neighbors (NP $= 85.1$) but leaves an indirect recovery rate of 24.5. The damage also tracks entanglement: STEREO's NP falls monotonically from 31.96 on `castle` ($\hat{\kappa} = 0.27$) to 1.81 on `dog` ($\hat{\kappa} = 0.89$; Section 5.3).

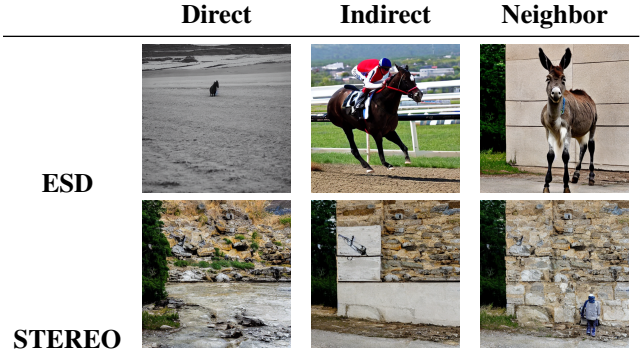


Figure 2: **Concept entanglement manifests as two coupled effects.** Each row erases horse and is probed with three prompt types: *Direct* (mentions "`horse`"), *Indirect* (uses correlated context like `jockey` but omits "`horse`"), and `Neighbor` (semantically similar concept `donkey`). ESD leaks horses through indirect prompts but preserves donkey; STEREO suppresses leakage but destroys donkey. We formalize this duality in Section 4.

4 The Robustness-Retention Trade-off: A Formal Impossibility Result

We now formalize the coupling observed in Section 3 as a provable lower bound on the trade-off between erasure robustness and concept preservation. The governing quantity is the *concept overlap coefficient* κ, which measures the fraction of the target concept's activation region shared with the remaining concepts.

4.1 Setup and Definitions

Let $\mathcal{D}_\theta$ be a pretrained text-to-image diffusion model with U-Net noise predictor ϵ_θ and text encoder $f_\psi : \mathcal{P} \to \mathbb{R}^d$, where $\mathcal{P}$ is the space of text prompts. For a prompt $p \in \mathcal{P}$, we denote its embedding

[3] We define entanglement formally in Section 4.1 based on activation regions in the model's cross-attention space.

by $\mathbf{e}_p = f_\psi(p) \in \mathbb{R}^d$, and write $p_\theta(\mathbf{x} \mid p)$ for the conditional distribution over images induced by $\mathcal{D}_\theta$. Let $\mathcal{D}_{\hat{\theta}}$ denote a concept-erased model (CEM) obtained by applying an unlearning procedure. We correspondingly write $\boldsymbol{\epsilon}_{\hat{\theta}}$ for the edited U-Net and $p_{\hat{\theta}}(\mathbf{x} \mid p)$ for the induced conditional distribution.

Concept activation regions. For a fixed probe layer ℓ^* and denoising timestep t^*, let $\mathbf{\Phi}_\theta^{\ell^*}(p)$ denote the mean-pooled cross-attention activation produced by prompt $p \in \mathcal{P}$. For a concept $c \in \mathcal{C}$, we define its *activation region* at layer ℓ^* as

$$\mathcal{R}_c(\ell^*) \;=\; \Big\{ \mathbf{\Phi}_\theta^{\ell^*}(p) \;\Big|\; p \in \mathcal{P},\; \mathbb{O}_{\mathcal{X}}(\mathbf{x}_\theta(p),\, c) = 1 \Big\} \;\subseteq\; \mathbb{R}^{d_{\ell^*}}, \tag{1}$$

where (i) $\mathbb{O}_{\mathcal{X}} : \mathcal{X} \times \mathcal{C} \to \{0, 1\}$ is an oracle indicating whether concept c is present in image x, (ii) $\mathbf{x}_\theta(p) \sim p_\theta(\cdot \mid p)$ denotes an image sampled from $\mathcal{D}_\theta$ conditioned on prompt p, and (iii) d_{ℓ^*} is the per-token channel dimension at layer at layer ℓ^*. We also define the closed ρ-neighborhood of $\mathcal{R}_c$ as:

$$\mathcal{R}_c^{(\rho)}(\ell^*) \;=\; \big\{ \mathbf{h} \in \mathbb{R}^{d_{\ell^*}} \;\big|\; \inf_{\mathbf{r} \in \mathcal{R}_c} \|\mathbf{h} - \mathbf{r}\|_2 \le \rho \big\} \tag{2}$$

When clear from context, we drop ℓ^* and simply write $\mathbf{\Phi}_\theta(p)$ and $\mathcal{R}_c^{(\rho)}$ for brevity. The key quantity governing the trade-off is the degree of representational overlap between the target concept and other concepts, defined as follows.

Definition 4.1. The *entanglement coefficient* of target concept c_u with fattening radius $\rho > 0$ is

$$\kappa(c_u,\, \rho) \;=\; \frac{\mathrm{vol}\Big(\mathcal{R}_{c_u}^{(\rho)} \cap \bigcup_{c' \in \mathcal{C} \setminus \{c_u\}} \mathcal{R}_{c'}^{(\rho)} \Big)}{\mathrm{vol}\Big(\mathcal{R}_{c_u}^{(\rho)} \Big)},$$

where $\mathrm{vol}(\cdot)$ denotes the n-dimensional volume of subsets of $\mathbb{R}^n$.

Intuitively, $\kappa \in [0, 1]$ measures what fraction of the target concept's representational footprint overlaps with that of other concepts in the model. When $\kappa = 0$, the concept is fully disentangled and can in principle be erased without collateral damage. When $\kappa > 0$, any edit that displaces activations in $\mathcal{R}_{c_u}$ will inevitably disturb overlapping portions of other concepts' activation regions. This formulation aggregates overlap across all concepts. In practice, however, the contribution to κ is dominated by concepts that occupy nearby regions of the activation space and distant concepts contribute negligibly to the intersection.

Quantifying entanglement. Figure 1 reports $\hat{\kappa}$, an empirical lower-bound estimator of the entanglement coefficient κ defined above, for five target concepts. Each $\hat{\kappa}(c_u)$ is computed as the fraction of activation samples from c_u that lie within a fattening radius ρ_{c_u} of any activation from a discovered neighbor concept; the full estimation procedure and per-target results are detailed in Appendix C.1.

Neighborhood construction. Neighbors are discovered automatically in the original activation space in two stages. For each target c_u we extract activations for ten semantically plausible candidates and three unrelated controls, and keep as neighbors $\mathcal{N}(c_u)$ the candidates whose centroid cosine distance to c_u falls below the 25th percentile of pairwise centroid distances in that pool (Definition C.1); the controls must fall outside $\mathcal{N}(c_u)$. $\hat{\kappa}$ is then computed from sample-level coverage, not from centroids: it is the fraction of target samples lying within the target's median intra-concept distance ρ_{c_u} of some neighbor sample. Because $\hat{\kappa}$ measures union coverage, enlarging $\mathcal{N}(c_u)$ can only weakly increase it, but cardinality does not determine it: `horse` and `castle` have seven and six neighbors yet $\hat{\kappa} = 0.81$ and 0.27. The estimate is also stable in practice: for `horse`, `mare` alone gives 0.78 and adding `pony` gives 0.81, after which five further neighbors change nothing; moving the percentile threshold from the 10th to the 50th changes $|\mathcal{N}|$ from 3 to 9 for `horse` and from 1 to 9 for `castle` while leaving $\hat{\kappa}$ at 0.81 and 0.27 (Appendix C.1).

We now define the two desiderata that any unlearning method must satisfy:

Definition 4.2 (ε-robust unlearning)**.** The CEM $\mathcal{D}_{\hat{\theta}}$ achieves *ε-robust unlearning* of concept c_u if, for all prompts $p \in \mathcal{P}$,

$$\Pr_{\mathbf{x} \sim p_{\hat{\theta}}(\cdot \mid p)} [\mathbb{O}_{\mathcal{X}}(\mathbf{x},\, c_u) = 1] \;\le\; \varepsilon. \tag{3}$$

See Appendix A for more details on the oracle.

Definition 4.3 (γ-concept preservation)**.** Let $\mathcal{P}_{c'} \subseteq \mathcal{P}$ denote the set of prompts associated with concept c', and let $\mathcal{P}_{-u}$ denote the *non-target prompt distribution*, defined as the uniform mixture $\mathcal{P}_{-u} = \frac{1}{|\mathcal{C}\setminus\{c_u\}|}\sum_{c'\in\mathcal{C}\setminus\{c_u\}} \mathcal{P}_{c'}$. The CEM $\mathcal{D}_{\hat{\theta}}$ achieves γ-*concept preservation* with respect to layer ℓ^* if

$$\mathbb{E}_{p\sim\mathcal{P}_{-u}}\left[\left\|\mathbf{\Phi}_{\hat{\theta}}^{\ell^*}(p) - \mathbf{\Phi}_{\theta}^{\ell^*}(p)\right\|_2\right] \;<\; \gamma. \tag{4}$$

4.2 Main Result: The Trade-off Lower Bound

In this section, we show that ε-robust unlearning and γ-concept preservation are fundamentally in tension. We begin by stating two mild regularity assumptions that formalize how perturbations in activation space affect generation.

Assumption 4.4 (Lipschitz continuity of generation)**.** For a model θ' and prompt p, let $g_c(\theta', p) = \Pr_{x\sim p_{\theta'}(\cdot|p)}[\mathbb{O}_{\mathcal{X}}(x, c) = 1]$. For each concept $c \in \mathcal{C}$ there is a constant L such that, for all prompts $p, p' \in \mathcal{P}$ and all $\theta', \theta'' \in \{\theta, \hat{\theta}\}$,

$$\left|g_c(\theta', p) - g_c(\theta'', p')\right| \le L\left\|\mathbf{\Phi}_{\theta'}(p) - \mathbf{\Phi}_{\theta''}(p')\right\|_2. \tag{5}$$

Interpretation. A single probe layer is not injective, and in a U-Net the decoder also receives encoder activations through skip connections, so we do not treat $\mathbf{\Phi}_\theta(p)$ as a sufficient statistic for generation. Assumption 4.4 is a smoothness condition along prompt- and model-dependent quantities: L is an effective constant of the full network that absorbs changes in the probe activation, the skip activations, and other hidden states, rather than a property of the probe layer in isolation. Appendix E estimates L for $\theta' = \theta'' = \theta$ over natural prompt variations; the proof uses the case $\theta' = \theta$, $\theta'' = \hat{\theta}$, $p' = p$.

Assumption 4.5 (Minimum displacement for erasure)**.** Any unlearning edit $\theta \mapsto \hat{\theta}$ that achieves ε-robust unlearning of c_u must displace activations within $\mathcal{R}_{c_u}$ by at least $\Delta > 0$:

$$\forall p \in \mathcal{P} \text{ such that } \mathbf{\Phi}_\theta(p) \in \mathcal{R}_{c_u}: \quad \|\mathbf{\Phi}_{\hat{\theta}}(p) - \mathbf{\Phi}_\theta(p)\|_2 \ge \Delta. \tag{6}$$

We empirically verify Assumption 4.4 across natural prompt variations and Assumption 4.5 across six unlearning methods erasing `horse`, including the gradient-based SalUn and TRUST; all measured pairs satisfy Lipschitz with a small empirical constant, and displacement propagates to other concepts through shared activation structure. Full details are in Appendices D and E.

Theorem 4.6 (Robustness-Retention trade-off)**.** *Let Assumptions 4.4 and 4.5 hold with constants L and Δ. Let $\rho > 0$ be the fattening radius from Definition 4.1. Suppose $\mathcal{D}_{\hat{\theta}}$ achieves ε-robust unlearning of c_u (Definition 4.2). Then the aggregate collateral displacement over non-target concepts satisfies*

$$\mathbb{E}_{p\sim\mathcal{P}_{-u}}\left[\|\mathbf{\Phi}_{\hat{\theta}}(p) - \mathbf{\Phi}_\theta(p)\|_2\right] \ge \kappa(c_u, \rho)\cdot\Delta. \tag{7}$$

Consequently, achieving γ-concept preservation(Definition 4.3) requires

$$\boxed{\gamma \;\ge\; \kappa(c_u, \rho)\cdot\frac{1-\varepsilon}{L}.} \tag{8}$$

Equivalently, when $\kappa(c_u, \rho) > 0$, achieving both ε-robustness and γ-concept preservation simultaneously requires

$$\varepsilon \;\ge\; 1 - \frac{L\gamma}{\kappa(c_u, \rho)}. \tag{9}$$

Proof sketch. Assumptions 4.4 and 4.5 together force every activation in $\mathcal{R}_{c_u}$ to be displaced by at least $\Delta \ge (1-\varepsilon)/L$ under any ε-robust edit. By Definition 4.1, a $\kappa(c_u, \rho)$-fraction of $\mathcal{R}_{c_u}^{(\rho)}$ lies in the support of $\mathcal{P}_{-u}$, so this displacement propagates to at least a $\kappa(c_u, \rho)$-fraction of $\mathcal{P}_{-u}$'s mass, yielding $\mathbb{E}_{p\sim\mathcal{P}_{-u}}[\|\Phi_{\hat{\theta}}(p) - \Phi_\theta(p)\|_2] \ge \kappa(c_u, \rho)(1-\varepsilon)/L$. Definition 4.3 then gives the bound on γ. The full proof including the regularity assumption is in Appendix G. □

Remark 4.7 (Scope: any unlearning mechanism). Theorem 4.6 makes no assumption about how $\hat{\theta}$ is obtained. $\mathcal{D}_{\hat{\theta}}$ is any model achieving ε-robust unlearning, so the bound applies equally to activation interventions, closed-form weight edits, gradient-based fine-tuning such as SalUn and TRUST, and inference-time methods, provided the resulting model displaces target activations as in Assumption 4.5. We verify this for gradient-based methods in Appendix D: TRUST, for example, displaces target activations by 15.82 ± 3.82, neighbors by 8.82 ± 4.88, and controls by only 3.10 ± 1.36.

4.3 Consequences and Interpretation

Limitations of simultaneous robust erasure and utility preservation. Based on Theorem 4.6, simultaneous perfect erasure and perfect retention is infeasible whenever concepts are entangled:

Corollary 4.8 (Impossibility result)**.** *If $\kappa(c_u, \rho) > 0$, then no unlearning method can simultaneously achieve $\varepsilon = 0$ (perfect erasure) and $\gamma = 0$ (perfect neighbor preservation).*

Proof. Setting $\varepsilon = 0$ in Eq. (8) gives $\gamma \geq \kappa(c_u, \rho)/L > 0$ under the stated conditions. □

The Pareto frontier. Eq. (9) defines a *feasibility boundary* in the (ε, γ) plane: no method can simultaneously achieve an erasure robustness below ε and a neighbor preservation error below γ unless the pair (ε, γ) lies above the curve $\gamma = \kappa \cdot \frac{1-\varepsilon}{L}$. Different unlearning methods correspond to different operating points on or above this curve: aggressive methods achieve small ε at the cost of large γ, occupying the upper-left of the frontier; conservative methods accept large ε to maintain small γ, occupying the lower-right. In Section 5, we verify that the empirically observed operating points across thirteen methods are consistent with this predicted boundary.

5 Experiments

We design our experiments to answer three questions, each tied to a component of our contribution:

Q1. Does the robustness-retention trade-off predicted by Theorem 4.6 manifest empirically across a broad set of unlearning methods? (Section 5.2)

Q2. Does the severity of the trade-off vary with the estimated entanglement coefficient $\hat{\kappa}$, as predicted by the theorem's dependence on κ? (Section 5.3)

Q3. Do the Lipschitz continuity (Assumption 4.4) and minimum displacement (Assumption 4.5) assumptions underlying Theorem 4.6 hold empirically? (Appendices D and E)

5.1 Experimental Setup

Base model and target concepts. We use CompVis Stable Diffusion v1.4 [Rombach et al., 2022] as the base model $\mathcal{D}_\theta$. Following [Gandikota et al., 2023], all generation uses 50 DDIM steps with guidance scale 7.5; full details in Appendix H. We select five target concepts spanning the range $\hat{\kappa} \in [0.27, 0.89]$ identified in Figure 1: four high-$\hat{\kappa}$ animal-subtype concepts (`horse`, `dog`, `cat`, `bear`, $\hat{\kappa} \in [0.77, 0.89]$) and one low-$\hat{\kappa}$ architectural concept (`castle`, $\hat{\kappa} = 0.27$). Each target is evaluated against its discovered concept neighborhood and three control concepts (Table 2); per-target prompt sets are detailed in Appendix I.1. To densify the intermediate range we add five targets, `snake`, `fish`, `tree`, `butterfly`, and `car` ($\hat{\kappa} = 0.37$ to 0.83), evaluated with four representative methods (Appendix J.6). Cross-architecture experiments on SDXL and FLUX are described in Appendix M.

Unlearning methods. We benchmark 13 methods across three intervention families: closed-form editing (UCE [Gandikota et al., 2024], RECE [Gong et al., 2024]); fine-tuning (ESD [Gandikota et al., 2023], SalUn [Fan et al., 2023], EDiff [Wu et al., 2024], CPE [Lee et al., 2025], TRUST [Kori et al., 2026], AdvUnlearn [Zhang et al., 2024], STEREO [Srivatsan et al., 2025]); and inference-time methods (SEOT [Li et al., 2024], SAeUron [Cywiński and Deja, 2025], SLD [Schramowski et al., 2023], SAFREE [Yoon et al., 2025]). AdvUnlearn and STEREO incorporate adversarial training to target the aggressive-erasure end of the trade-off. Method-specific configurations are detailed in Appendix H.

Experimental data and sources. We use no existing labeled image test set. All images are generated by Stable Diffusion v1.4 or its unlearned variants from four controlled prompt families per target: direct prompts naming the target, indirect recovery prompts built from correlated context that excludes the target name, neighbor prompts naming concepts in $\mathcal{N}(c)$, and control prompts naming unrelated concepts. Correlated context terms are mined from target-matched LAION caption text [Schuhmann et al., 2022]; LAION images are not used. For each target we generate around 200 images per prompt family (exact counts in Appendix H.3), all with 50 DDIM steps, guidance scale 7.5, and fixed seeds shared across the original and unlearned models. Prompt files, seeds, the concept vocabulary, and discovered neighborhoods are released with the code (Appendices H and I).

Method	Type	Erasure		Retention			DamageGap↓
		UA↑	IRR↓	IRA↑	NP↑	CP↑	
Original	–	0.0 ± 0.0	58.7 ± 38.1	70.45 ± 12.6	100.0 ± 0.0	100.0 ± 0.0	–
UCE	Closed-form	80.6 ± 18.4	10.5 ± 23.4	67.1 ± 12.4	88.6 ± 25.1	99.3 ± 3.1	10.7 ± 25.4
RECE	Closed-form	99.4 ± 0.9	3.0 ± 4.0	43.3 ± 12.1	56.1 ± 10.6	93.5 ± 2.0	37.4 ± 10.2
ESD	FT	80.7 ± 11.1	20.9 ± 14.2	59.2 ± 12.7	80.6 ± 13.5	97.0 ± 1.6	16.4 ± 14.5
SalUn	FT	93.7 ± 5.1	12.8 ± 11.5	46.5 ± 8.5	58.4 ± 13.0	90.2 ± 8.9	31.8 ± 13.1
EDiff	FT	79.4 ± 12.2	17.4 ± 12.4	55.6 ± 12.3	73.4 ± 12.1	95.8 ± 3.2	22.4 ± 10.9
CPE	FT	99.6 ± 0.9	4.7 ± 9.4	53.4 ± 12.4	71.9 ± 18.1	99.6 ± 2.6	27.7 ± 19.3
TRUST	FT	94.2 ± 2.2	6.5 ± 3.5	53.2 ± 13.7	76.4 ± 9.1	98.0 ± 5.0	21.6 ± 12.2
AdvUnlearn	Robust FT	98.5 ± 1.7	9.0 ± 11.5	55.4 ± 13.9	71.5 ± 7.5	99.4 ± 0.9	27.9 ± 7.3
STEREO	Robust FT	98.5 ± 3.1	3.5 ± 3.5	19.7 ± 10.0	18.0 ± 12.0	70.8 ± 27.4	52.8 ± 22.7
SEOT	Inf.	82.6 ± 13.2	24.5 ± 13.8	59.3 ± 15.0	85.1 ± 13.8	98.3 ± 1.3	13.6 ± 13.0
SAeUron†	Inf.	78.4 ± 5.5	27.6 ± 12.7	42.1 ± 31.8	59.0 ± 37.8	89.2 ± 10.8	30.2 ± 27.2
SLD	Inf.	99.6 ± 0.9	0.5 ± 0.9	35.0 ± 8.1	58.8 ± 16.8	89.1 ± 4.1	30.3 ± 16.3
SAFREE	Inf.	91.4 ± 11.4	8.4 ± 10.2	39.9 ± 8.6	64.7 ± 16.0	85.7 ± 12.2	21.0 ± 19.4

Table 1: **Averaged evaluation of unlearning methods on the robustness-retention trade-off.** Results are averaged over the five target concepts (`dog`, `bear`, `horse`, `cat`, `castle`) with $\hat{\kappa} \in [0.27, 0.89]$. Cells report mean $\pm$ sample standard deviation across concepts. UA: Unlearning Accuracy; IRR: Indirect Recovery Rate; IRA/CP: In-domain/Cross-domain Retain Accuracy; NP: Neighbor Preservation (Eq. (27)); DamageGap: composite retention-damage summary. The *Original* row reports baseline rates for the pretrained $\mathcal{D}_\theta$. A few runs use non-standard image counts; see the per-concept table captions in Appendix J. †SAeUron is averaged over four concepts (`dog`, `bear`, `horse`, `cat`); its released sparse autoencoder features do not cover `castle` (see Appendix J.5).

Evaluation. We report six metrics spanning the two axes of Theorem 4.6 (full definitions in Appendix I.2). Every image is classified by `openai/clip-vit-base-patch32` over a fixed 65-concept vocabulary (Appendix I.3); each concept's text representation averages normalized CLIP embeddings of four templates, and the predicted label $\hat{y}(x)$ is the concept with the highest cosine similarity. With $\mathcal{X}_{\text{dir}}(c)$ and $\mathcal{X}_{\text{ind}}(c)$ the images from direct and indirect recovery prompts,

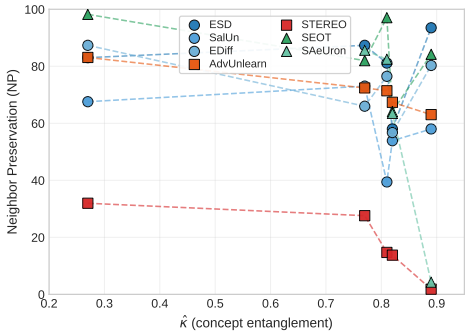
$$\text{UA}(c) = 1 - \frac{1}{|\mathcal{X}_{\text{dir}}(c)|} \sum_{x \in \mathcal{X}_{\text{dir}}(c)} \mathbf{1}[\hat{y}(x) = c], \qquad \text{IRR}(c) = \frac{1}{|\mathcal{X}_{\text{ind}}(c)|} \sum_{x \in \mathcal{X}_{\text{ind}}(c)} \mathbf{1}[\hat{y}(x) = c]. \quad (10)$$

IRR is our worst-case proxy for ε. On labeled images of the five targets, the classifier attains a micro-averaged TPR of 97.0% and FPR of 0.55% (Appendix I.4). We additionally report In-domain Retain Accuracy (IRA), the fraction of neighbor and control images classified as the concept they were meant to show. *Retention metrics* quantify collateral damage. *Neighbor Preservation* (NP), is the classifier accuracy on generations from prompts naming semantically related concepts (e.g., `donkey`, `pony` when erasing `horse`). We additionally report *Control Preservation* (CP) on unrelated concepts and $\text{DamageGap} = \text{CP} - \text{NP}$, which isolates neighbor-selective damage from uniform degradation. All metrics are reported both absolutely and as relative changes from $\mathcal{D}_\theta$ to enable cross-concept comparison.

5.2 The Trade-off is Real and Consistent

We first test whether the robustness-retention trade-off predicted by Theorem 4.6 manifests empirically across methods. Table 1 reports each method's performance averaged over the five target concepts. Per-concept breakdowns are provided in Appendix J

No method escapes the trade-off. Table 1 reveals a consistent pattern across thirteen methods: no method simultaneously achieves high erasure strength (high UA, low IRR) and high neighbor retention. Averaged over concepts, no method reaches both UA > 90 and NP > 80. The most balanced operating points come from methods designed to preserve non-target concepts, yet they remain on the frontier: TRUST reaches UA 94.2 at NP 76.4, and UCE attains the highest NP (88.6) but with weaker and highly variable erasure (UA 80.6 ± 18.4). Methods that achieve near-complete



Concept	$\hat{\kappa}$	NP↑	IRA-ratio↑
`castle`	0.27	31.96	0.55
`cat`	0.77	27.60	0.32
`horse`	0.81	14.71	0.29
`bear`	0.82	13.75	0.25
`dog`	0.89	1.81	0.06
r_s		-1.0	-1.0

Figure 3: **Neighbor preservation scales inversely with $\hat{\kappa}$.** *Left:* NP for each (method, concept) pair among the seven original methods; dashed lines connect each method's points in order of $\hat{\kappa}$ and serve as visual guides, not fitted trends. *Right:* STEREO's NP and in-domain retention ratio across the five concepts ($r_s = -1.0$ for both).

erasure (RECE, CPE, SLD, STEREO; UA ≥ 99.4) retain NP between 18.0 and 71.9. Mechanism family does not determine the operating point: the inference-time method SLD attains the lowest IRR (0.5) at NP 58.8, whereas SEOT, also an inference-time method, preserves neighbors (NP 85.1) but leaks (IRR 24.5). Algorithm design thus determines where a method lies on the frontier, while concept entanglement constrains the frontier it faces, consistent with Corollary 4.8. Appendix F interprets these methods as operating points induced by their effective displacement Δ, and Figure 8 plots every per-concept operating point of the seven original methods.

The trade-off is structural, not a measurement artifact. The DamageGap column spans more than 40 points across methods (from UCE at 10.7 to STEREO at 52.8), and the NP column spans more than 70 points (from STEREO at 18.0 to UCE at 88.6). If the trade-off were driven by noise or evaluation instability, we would expect method-level differences to be dominated by concept-level variation; instead, method identity is the primary determinant of operating region, with concept identity modulating *severity* of the trade-off at each operating point. We characterize this concept-level scaling next (Section 5.3) and provide a detailed per-concept view for `horse` in Appendix J.

5.3 Semantic Density Governs Trade-off Severity

Theorem 4.6 predicts that trade-off severity scales with the entanglement coefficient κ: target concepts whose activation regions overlap substantially with other concepts should suffer greater collateral damage under robust erasure. We test this across five target concepts with estimated $\hat{\kappa}$ span from 0.27 (`castle`) to 0.89 (`dog`); see Figure 1 and Appendix C for the estimation procedure.

$\hat{\kappa}$ predicts collateral damage on neighbors. Figure 3 plots NP against $\hat{\kappa}$ for the available method-concept pairs among the seven original methods, and the table on the right reports STEREO's NP and in-domain retention ratio ordered by ascending $\hat{\kappa}$. Both decrease monotonically as $\hat{\kappa}$ increases, consistent with the linear κ-dependence in Eq. (8). Figure 3 reveals three patterns: robust fine-tuning methods (red squares) exhibit the steepest $\hat{\kappa}$-dependence, with STEREO tracking $\hat{\kappa}$ monotonically and AdvUnlearn similarly decreasing; inference-time methods (green triangles) show only weak $\hat{\kappa}$-dependence because their effective displacement Δ is too small for the κ-scaling to dominate; standard fine-tuning methods (blue circles) occupy the middle band. This stratification reflects a structural principle: $\hat{\kappa}$ governs the lower bound on collateral damage, while the method's effective Δ determines how close it operates to that bound.

Direct test of Eq. (8). Eq. (8) predicts that, at matched erasure strength, a larger κ forces a larger collateral displacement γ. STEREO provides this controlled comparison: it reaches UA ≥ 99.5 on four of five targets and 93.0 on `cat`. Measured directly in activation space (Appendix K, Table 14), its mean neighbor displacement rises from 75.9 on `castle` ($\hat{\kappa} = 0.27$) to 234.2, 233.4, 240.9, and 257.0 on `cat`, `horse`, `bear`, and `dog` ($r_s = 0.90$). All five fine-tuning methods show a positive correlation, four of them at $r_s \geq 0.70$, and ESD is strictly monotone ($r_s = 1.00$). Inverting Eq. (8) at each STEREO operating point yields $L_{\text{tight}} \in [0.0032, 0.0035]$ across concepts, a 7.8% spread, consistent with a single local Lipschitz constant for a method operating near the bound. Table 1

therefore establishes the trade-off, while this analysis tests the κ-dependence of the bound in the units in which it is stated; NP serves as a probability-level proxy for γ.

Denser coverage of $\hat{\kappa}$. To test whether the trend depends on `castle` alone or on an animal versus architecture split, we add `snake`, `fish`, `tree`, `butterfly`, and `car` ($\hat{\kappa} = 0.37, 0.52, 0.63, 0.66, 0.83$) and evaluate EDiff, ESD, SalUn, and STEREO (Appendix J.6). Method-averaged NP decreases monotonically with $\hat{\kappa}$, from 65.0 to 58.8 ($r_s = -1.0$), and every within-method correlation is negative (EDiff -0.4, ESD -0.7, SalUn -0.2, STEREO -0.7). With five additional targets these correlations are descriptive, not individually significant; they show that the direction of the effect persists across animal, plant, and vehicle concepts, including a low-$\hat{\kappa}$ animal and a high-$\hat{\kappa}$ vehicle. The $\hat{\kappa}$ ordering is stable across probe layers ($r_s \geq 0.80$) and denoising timesteps ($r_s = 0.93$ and 0.96 at steps 10 and 40 relative to step 25), although absolute values vary (Appendix C.2). A discussion of why $\hat{\kappa}$ reflects activation overlap rather than semantic category is in Appendix C.

Beyond Stable Diffusion v1.4. On SDXL, whose cross-attention layers admit $\hat{\kappa}$, ESD's neighbor preservation decreases monotonically with entanglement: NP $= 90.9$, 85.0, and 77.1 for `cat`, `castle`, and `dog` ($\hat{\kappa}_{\text{SDXL}} = 0.65, 0.72, 0.92$; $r_s = -1.0$ over three concepts). The most entangled target is also the least erased (UA $= 77$ for `dog` versus 99 for `cat`), so the ordering is not driven by erasure strength. Absolute $\hat{\kappa}$ shifts across architectures (`castle` is 0.27 on SD v1.4 and 0.72 on SDXL), but $\hat{\kappa}$ and observed damage agree within each architecture. On the MMDiT model FLUX, joint text-image attention leaves no cross-attention block analogous to our probe, and a proxy estimator saturates near 1 even for unrelated control pairs, so κ-scaling on MMDiT remains untested. The phenomenon itself recurs: ESD on FLUX reduces NP to 35.6 to 40.5 on `dog`, `cat`, and `castle` while preserving unrelated controls (CP $= 95.3$ to 98.9), i.e., neighbor-selective rather than global degradation (Appendix M).

6 Conclusion

We show that the tension between robust concept erasure and utility preservation in diffusion models is a structural consequence of representation geometry, not merely a limitation of existing methods. Theorem 4.6 establishes a κ-scaled lower bound on collateral damage, and our experiments show that existing methods, including those designed to preserve non-target concepts, occupy different points on the resulting robustness-retention frontier, and the frontier becomes steeper as $\hat{\kappa}$ increases.

Implications. Three consequences follow. First, $\hat{\kappa}$ is a pre-deployment diagnostic: estimated from base-model activations alone, it indicates which targets admit clean erasure and which require accepting collateral damage, before any unlearning is run (Appendix J.7). Second, benchmarks should report erasure and retention jointly, on neighbor sets selected by activation overlap, because reporting either axis alone hides where a method sits on the frontier. Third, method design should be κ-aware: rather than suppressing broad adversarial subspaces, methods should restrict displacement to the directions needed for erasure.

7 Limitations

Scope of empirical evaluation. All thirteen methods are evaluated on five targets on Stable Diffusion v1.4; five further targets are evaluated with four methods. Cross-architecture evidence is limited: SDXL covers three concepts with one method, and on FLUX we can show neighbor-selective damage but cannot resolve κ, so κ-scaling on MMDiT remains open. Our targets are object concepts; artistic styles, identities, and harmful-content concepts remain untested.

Theory and $\hat{\kappa}$ estimation. Theorem 4.6 applies to any unlearning mechanism, but its constants are effective ones: L captures the full network, including skip connections, and is estimated only for the prompt variations of Appendix E. $\hat{\kappa}$ is an empirical lower-bound proxy for κ whose absolute value depends on the probe layer, the denoising timestep (e.g., $\hat{\kappa}(\texttt{castle}) = 0.77$ at step 10), and the candidate pool; our claims rely on its ranking, which is stable across the settings we test. The text-space comparison is not a causal decomposition of where entanglement arises.

Acknowledgments

This work used Delta computing resources at National Center for Supercomputing Applications through allocation CIS240341 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program Boerner et al. [2023], which is supported by U.S. National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296. This work was supported by National Science Foundation Award #2312561. Varun Chandrasekaran gratefully acknowledges support from Google through the Google Research Scholar Award.

References

Timothy J. Boerner, Stephen Deems, Thomas R. Furlani, Shelley L. Knuth, and John Towns. Access: Advancing innovation: Nsf's advanced cyberinfrastructure coordination ecosystem: Services & support. In *Practice and Experience in Advanced Research Computing 2023: Computing for the Common Good*, PEARC '23, page 173–176, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9781450399852. doi: 10.1145/3569951.3597559. URL https://doi.org/10.1145/3569951.3597559.

Anh Bui, Long Vuong, Khanh Doan, Trung Le, Paul Montague, Tamas Abraham, and Dinh Phung. Erasing undesirable concepts in diffusion models with adversarial preservation. *arXiv preprint arXiv:2410.15618*, 2024.

Bartosz Cywiński and Kamil Deja. Saeuron: Interpretable concept unlearning in diffusion models with sparse autoencoders. *arXiv preprint arXiv:2501.18052*, 2025.

Chongyu Fan, Jiancheng Liu, Yihua Zhang, Eric Wong, Dennis Wei, and Sijia Liu. Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. *arXiv preprint arXiv:2310.12508*, 2023.

Rohit Gandikota, Joanna Materzynska, Jaden Fiotto-Kaufman, and David Bau. Erasing concepts from diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2426–2436, 2023.

Rohit Gandikota, Hadas Orgad, Yonatan Belinkov, Joanna Materzyńska, and David Bau. Unified concept editing in diffusion models. In *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5099–5108. IEEE, 2024.

Hongcheng Gao, Tianyu Pang, Chao Du, Taihang Hu, Zhijie Deng, and Min Lin. Meta-unlearning on diffusion models: Preventing relearning unlearned concepts. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2131–2141. IEEE, 2025.

Chao Gong, Kai Chen, Zhipeng Wei, Jingjing Chen, and Yu-Gang Jiang. Reliable and efficient concept erasure of text-to-image diffusion models. In *European Conference on Computer Vision*, pages 73–88. Springer, 2024.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Chi-Pin Huang, Kai-Po Chang, Chung-Ting Tsai, Yung-Hsuan Lai, Fu-En Yang, and Yu-Chiang Frank Wang. Receler: Reliable concept erasing of text-to-image diffusion models via lightweight erasers. In *European Conference on Computer Vision*, pages 360–376. Springer, 2024.

Avinash Kori, Francesca Toni, Soteris Demetriou, et al. Selective fine-tuning for targeted and robust concept unlearning. *arXiv preprint arXiv:2602.07919*, 2026.

Byung Hyun Lee, Sungjin Lim, Seunggyu Lee, Dong Un Kang, and Se Young Chun. Concept pinpoint eraser for text-to-image diffusion models via residual attention gate. In *International Conference on Learning Representations*, volume 2025, pages 99119–99158, 2025.

Senmao Li, Joost van de Weijer, Taihang Hu, Fahad Shahbaz Khan, Qibin Hou, Yaxing Wang, and Jian Yang. Get what you want, not what you don't: Image content suppression for text-to-image diffusion models. *arXiv preprint arXiv:2402.05375*, 2024.

Shilin Lu, Zilan Wang, Leyang Li, Yanzhu Liu, and Adams Wai-Kin Kong. Mace: Mass concept erasure in diffusion models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6430–6440. IEEE, 2024.

Mengyao Lyu, Yuhong Yang, Haiwen Hong, Hui Chen, Xuan Jin, Yuan He, Hui Xue, Jungong Han, and Guiguang Ding. One-dimensional adapter to rule them all: Concepts diffusion models and erasing applications. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7559–7568, 2024.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

Patrick Schramowski, Manuel Brack, Björn Deiseroth, and Kristian Kersting. Safe latent diffusion: Mitigating inappropriate degeneration in diffusion models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22522–22531. IEEE, 2023.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems*, 35:25278–25294, 2022.

Koushik Srivatsan, Fahad Shamshad, Muzammal Naseer, Vishal M Patel, and Karthik Nandakumar. Stereo: A two-stage framework for adversarially robust concept erasing from text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23765–23774, 2025.

Zihao Wang, Yuxiang Wei, Fan Li, Renjing Pei, Hang Xu, and Wangmeng Zuo. Ace: Anti-editing concept erasure in text-to-image models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23505–23515, 2025.

Jing Wu and Mehrtash Harandi. Scissorhands: Scrub data influence via connection sensitivity in networks. In *European Conference on Computer Vision*, pages 367–384. Springer, 2024.

Jing Wu, Trung Le, Munawar Hayat, and Mehrtash Harandi. Erasediff: Erasing data influence in diffusion models. *arXiv preprint arXiv:2401.05779*, 2024.

Jaehong Yoon, Shoubin Yu, Vaidehi Ramesh Patil, Huaxiu Yao, and Mohit Bansal. Safree: Training-free and adaptive guard for safe text-to-image and video generation. In *International Conference on Learning Representations*, volume 2025, pages 56439–56465, 2025.

Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? *arXiv preprint arXiv:2210.01936*, 2022.

Arman Zarei, Samyadeep Basu, Keivan Rezaei, Zihao Lin, Sayan Nag, and Soheil Feizi. Localizing knowledge in diffusion transformers. *Advances in Neural Information Processing Systems*, 38: 14584–14612, 2026.

Yimeng Zhang, Xin Chen, Jinghan Jia, Yihua Zhang, Chongyu Fan, Jiancheng Liu, Mingyi Hong, Ke Ding, and Sijia Liu. Defensive unlearning with adversarial training for robust concept erasure in diffusion models. *arXiv preprint arXiv:2405.15234*, 2024.

Appendix

A Formal Definition of Concept Activation Regions

A.1 Background: latent diffusion models

We briefly introduce the latent diffusion notation used throughout the paper, since our analysis of concept activation regions and unlearning-induced perturbations is defined over the model's latent, text-conditioned denoising process.

Text-to-image latent diffusion models. We consider a text-to-image latent diffusion model (LDM) of the form introduced in Stable Diffusion [Rombach et al., 2022]. An input image $\mathbf{x} \in \mathcal{X}$ is first mapped by a pretrained variational autoencoder encoder E into a latent representation $\mathbf{z} = E(x) \in \mathcal{Z}$, and reconstructed by a decoder $V : \mathcal{Z} \to \mathcal{X}$.

Given a latent $\mathbf{z}$, the forward diffusion process corrupts it with Gaussian noise to obtain $\mathbf{z}_t$ at timestep $t \in \{1, \dots, T\}$. A time-conditioned U-Net $\boldsymbol{\epsilon}_\theta$ is then trained to predict the noise $\boldsymbol{\epsilon}$ added to $\mathbf{z}_t$. In text-conditioned generation, a prompt $p \in \mathcal{P}$ is encoded by a text encoder f_ψ into text features $\mathbf{e}_p = f_\psi(p)$, and these features condition the U-Net through cross-attention layers. The standard latent diffusion objective is

$$\mathcal{L}_{\text{LDM}} = \mathbb{E}_{x,\epsilon,t}\left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t, \mathbf{e}_p)\|_2^2\right], \tag{11}$$

where $\epsilon \sim \mathcal{N}(0, I)$ At inference time, generation starts from Gaussian noise and iteratively denoises toward a clean latent, which is finally decoded into an image.

A.2 Concept activation regions

This appendix provides the formal construction of the concept activation function $\boldsymbol{\Phi}_\theta$ and activation region $\mathcal{R}_c$ used throughout Section 4. The compressed version in the main text specifies these informally; the definition here is the formal object used in the proofs of Theorem 4.6.

Definition A.1 (Concept activation function and region)**.** Fix a UNet layer ℓ^* and denoising timestep $t^* \in \{1, \dots, T\}$. Let $p \in \mathcal{P}$ denote a text prompt and $\mathbf{e}_p = f_\psi(p)$ be its text embedding. The *concept activation function* of the pretrained model is

$$\boldsymbol{\Phi}_{\theta,\ell^*,t^*}(p) \;=\; \mathrm{A}_\theta(\mathbf{e}_p, \ell^*, t^*) \in \mathcal{H}, \tag{12}$$

where $\mathrm{A}_\theta(\mathbf{e}_p, \ell^*, t^*)$ denotes the cross-attention output of ϵ_θ at layer ℓ^* and timestep t^*, and $\mathcal{H} := \mathbb{R}^{d_{\ell^*}}$ is the corresponding activation space. Throughout, we suppress the dependence on ℓ^* and t^* and write $\boldsymbol{\Phi}_\theta(p)$ for brevity. Analogously, $\boldsymbol{\Phi}_{\hat{\theta}}(p)$ denotes the activation function of the CEM $\mathcal{D}_{\hat{\theta}}$.

For a concept $c \in \mathcal{C}$, its *activation region* is

$$\mathcal{R}_c \;=\; \left\{ \boldsymbol{\Phi}_\theta(p) \;\middle|\; p \in \mathcal{P},\ \mathbb{O}_{\mathcal{X}}(\mathbf{x}_\theta(p),\, c) = 1 \right\} \;\subseteq\; \mathcal{H}, \tag{13}$$

where $\mathbb{O}_{\mathcal{X}} : \mathcal{X} \times \mathcal{C} \to \{0, 1\}$ is an oracle indicating whether concept c is present in image $\mathbf{x}$, and $\mathbf{x}_\theta(p) \sim p_\theta(\cdot \mid p)$ denotes an image sampled from $\mathcal{D}_\theta$ conditioned on prompt p.

On the oracle classifier. The predicate $c \in \text{image}(\mathbf{x})$ is specified as an oracle classifier rather than a specific model because our results are independent of the particular choice. Any reasonable concept detector induces a concept region $\mathcal{R}_c$, and the tradeoff in Theorem 4.6 holds with respect to whichever classifier is fixed. In Section 5, we instantiate this oracle with a CLIP-based classifier, detailed in Appendix I.

B Empirical Validation of Concept Activation Regions

This section provides empirical support for Definition A.1 by showing that the cross-attention activations used in our formalization form coherent geometric clusters in practice.

Setup. To verify that concept activation regions are well-defined empirical objects, we extract cross-attention activations from Stable Diffusion v1.4 at the mid-block layer (`mid_block.attentions.0`) at step t^*=25 of 50 DDIM steps for 10 concepts, each prompted with 50 shared templates differing only in the concept word. `horse`, `pony`, `donkey`, `deer`, `dog`, `cat`, `bear`, `car`, `truck`, and `castle`. This visualization pool is chosen to be illustrative rather than comprehensive: it deliberately spans three semantically distinct domains (animals, vehicles, architecture) so that the qualitative geometry of concept regions (semantically related concepts clustering while distinct domains separate) is visually apparent. Several concepts in this pool (`deer`, `car`, `truck`) are not part of the evaluation vocabulary in Table 5; they appear only to anchor the visualization's domain diversity. Conversely, $\hat{\kappa}$-estimation and all CLIP-based evaluation metrics use the full 65-concept pool of Table 5, of which the visualization pool is not a strict subset. For each concept, we generate 50 prompts using a shared set of templates that differ only in the concept word, so that differences in activation primarily reflect the concept token rather than unrelated prompt variation.

For each prompt p, we compute the activation

$$\mathbf{\Phi}_\theta(p) \;=\; \mathrm{A}_\theta(\mathbf{e}_p, \ell^*, t^*) \in \mathcal{H}, \tag{14}$$

flatten the resulting cross-attention tensor into a vector in $\mathcal{H}$, and project all activations to two dimensions using UMAP for visualization.

Observations. Figure 1 shows that activations induced by prompts containing the same concept form concentrated local clusters, rather than being arbitrarily scattered in activation space. This supports the view that each concept corresponds to a coherent region in the model's internal representation space.

The figure also reveals non-uniform distances between concepts. Some concepts occupy nearby regions (for example, `horse`, `pony`, and `donkey` form a tight cluster), while others, such as `castle`, are clearly separated from the animal concepts. Importantly, we do not use these visualizations to infer semantic relatedness from scratch. Rather, the figure shows that the activation geometry is structured and that concepts commonly regarded as related in the data domain tend to induce nearby activations under shared prompt templates.

These results do not by themselves prove overlap in the strict set-theoretic sense used later in Section 4. Instead, they provide empirical evidence for the weaker claim needed to motivate our definitions: concept-conditioned activations are localized, stable enough to visualize, and often organized into neighboring regions rather than isolated points. This makes it meaningful to model a concept through an activation region $\mathcal{R}_c$ and to reason about neighborhood structure in the induced activation space.

Choice of probe layer and timestep. We probe the cross-attention output at `mid_block.attentions.0` at denoising step $t^* = 25$ of 50 throughout Appendices B to E, and at `up_blocks.1.attentions.1` (same timestep) for the direct displacement measurements in Appendix K. The probe layer ℓ^* is a parameter of our framework (Definition A.1), not a property of the model, and Theorem 4.6 holds for any chosen (ℓ^*, t^*) at which Assumptions 4.4 and 4.5 are satisfied. We selected mid-block as the default probe based on a sensitivity analysis (Appendix C.2) over six candidate cross-attention layers spanning the U-Net depth: the $\hat{\kappa}$ ordering across our five target concepts is stable across all probed layers ($r_s \geq 0.80$ vs. mid-block), but mid-block produces the cleanest bimodal separation between high-$\hat{\kappa}$ and low-$\hat{\kappa}$ concepts and the tightest within-cluster spread among the animal subtypes. Step $t^* = 25$ (the midpoint of the DDIM trajectory) corresponds to the regime in which large-scale layout and object identity are committed in latent diffusion models. The up-block layer in Appendix K was selected because it lies downstream of the cross-attention edits applied by fine-tuning methods and therefore reflects the cumulative effect of those edits; Appendix C.2 confirms that the $\hat{\kappa}$ ranking at this layer is consistent with mid-block.

C Estimation of the Entanglement Coefficient $\hat{\kappa}$

The entanglement coefficient $\kappa(c_u, \rho)$ defined in Definition 4.1 involves the volume of high-dimensional activation regions, which is intractable to compute directly. We therefore estimate κ via a sample-based procedure that restricts the overlap measurement to a discovered *semantic neighborhood* $\mathcal{N}(c_u) \subseteq \mathcal{C} \setminus \{c_u\}$ rather than the full union over remaining concepts. Because $\mathcal{N}(c_u)$ is a subset of the union appearing in Definition 4.1, this neighborhood-restricted estimate

is a *lower bound* on $\kappa(c_u, \rho)$. We argue, and verify in Appendix C.1, that this lower bound is tight in practice: across the five targets, the activation overlap with non-neighbors is negligible, so the union over $\mathcal{C} \setminus \{c_u\}$ is dominated by the union over $\mathcal{N}(c_u)$. Throughout this section we work with cosine distance $d_{\cos}(\mathbf{a}, \mathbf{b}) = 1 - \langle \mathbf{a}, \mathbf{b} \rangle / (\|\mathbf{a}\| \|\mathbf{b}\|)$, which is the standard choice in representation analysis: it is invariant to activation magnitude (which is dominated by token-position effects under mean-pooling) and captures directional similarity, the property most closely tied to concept identity. The centroid-distance matrix used to motivate neighborhood structure in Section 4 (Figure 4) uses the same metric, ensuring internal consistency between our motivation and our formalization.

On the relationship between cosine and ℓ_2 metrics. Theorem 4.6 bounds the ℓ_2 displacement $\|\mathbf{\Phi}_{\hat{\theta}}(p) - \mathbf{\Phi}_{\theta}(p)\|_2$, since this quantity measures a physical perturbation to the activation vector. Our κ estimator, in contrast, uses cosine distance to characterize region membership. These two choices are compatible: for activations normalized to the unit sphere, $\|\mathbf{a} - \mathbf{b}\|_2^2 = 2 \cdot d_{\cos}(\mathbf{a}, \mathbf{b})$, so an L-Lipschitz function under the ℓ_2 metric satisfies $|f(a) - f(b)| \leq L\sqrt{2\, d_{\cos}(a, b)}$ on the unit sphere, i.e., it is Hölder continuous with exponent $1/2$ in cosine distance. The theorem is stated and proved entirely in ℓ_2; cosine distance is used only to decide region membership in the estimator, and since both metrics induce the same fattened regions up to a monotone re-parameterization of ρ, this choice does not affect the bound.

Activation extraction. For each concept c, we generate images from 50 shared prompt templates (differing only in the concept word) using the base model (Stable Diffusion v1.4) with DDIM sampling (50 steps, guidance scale 7.5). At step t^*=25, we extract the cross-attention output at `mid_block.attentions.0`, take the text-conditioned component, and mean-pool over spatial dimensions to obtain a single activation vector $\mathbf{\Phi}_{\theta}(p) \in \mathbb{R}^d$ per prompt.

Neighborhood discovery. We restrict the overlap measurement in Definition 4.1 to a finite set of candidate concepts that are close to c_u in activation space. We formalize this set as follows.

Definition C.1 (Semantic neighborhood)**.** For each target c_u, let $\mathcal{C}_{c_u} = \{c_u\} \cup \mathcal{C}_{c_u}^{\text{cand}} \cup \mathcal{C}_{c_u}^{\text{ctrl}}$ be a target-specific pool, where $\mathcal{C}_{c_u}^{\text{cand}}$ contains ten semantically plausible candidate neighbors of c_u and $\mathcal{C}_{c_u}^{\text{ctrl}}$ contains three unrelated control concepts. For each $c \in \mathcal{C}_{c_u}$, let $\bar{\mathbf{a}}_c = \frac{1}{|\mathcal{S}_c|} \sum_{\mathbf{a} \in \mathcal{S}_c} \mathbf{a}$ be the centroid of its activation samples. Let δ_{c_u} be the 25th percentile of all pairwise centroid cosine distances within $\mathcal{C}_{c_u}$. The semantic neighborhood of c_u is

$$\mathcal{N}(c_u) = \left\{ c \in \mathcal{C}_{c_u}^{\text{cand}} : d_{\cos}(\bar{\mathbf{a}}_{c_u}, \bar{\mathbf{a}}_c) < \delta_{c_u} \right\}, \tag{15}$$

and every control is required to fall outside $\mathcal{N}(c_u)$.

We emphasize that δ_{c_u} and $\mathcal{N}(c_u)$ are *estimation-side* constructs: they appear only in the empirical estimator $\hat{\kappa}$ below and play no role in the theoretical bound of Theorem 4.6, which is stated entirely in terms of $\kappa(c_u, \rho)$. Because $\mathcal{N}(c_u) \subseteq \mathcal{C} \setminus \{c_u\}$, restricting the overlap computation to $\mathcal{N}(c_u)$ yields a lower bound on the true κ. The discovered neighborhoods are reported in Table 2.

Overlap estimation. For a target concept c_u with discovered neighbors $\mathcal{N}(c_u)$, we estimate $\hat{\kappa}$ as the fraction of c_u's activation samples that fall within radius ρ of any neighbor's activation samples:

$$\hat{\kappa}(c_u) \;=\; \frac{1}{|\mathcal{S}_{c_u}|} \sum_{\mathbf{a} \in \mathcal{S}_{c_u}} \mathbf{1}\left[\min_{c' \in \mathcal{N}(c_u)} \min_{\mathbf{b} \in \mathcal{S}_{c'}} d_{\cos}(\mathbf{a}, \mathbf{b}) \leq \rho \right], \tag{16}$$

where $\mathcal{S}_c = \{\mathbf{\Phi}_{\theta}(p_i)\}_{i=1}^{50}$ denotes the set of activation samples for concept c, and $d_{\cos}$ is the cosine distance. The fattening radius ρ_{c_u} is set to the median intra-concept pairwise cosine distance within c_u:

$$\rho_{c_u} \;=\; \text{median}\{ d_{\cos}(\mathbf{a}, \mathbf{a}') \mid \mathbf{a}, \mathbf{a}' \in \mathcal{S}_{c_u},\ \mathbf{a} \neq \mathbf{a}' \}\,. \tag{17}$$

This choice calibrates the overlap radius to the natural spread of the target concept's activation region: a neighbor activation qualifies as overlapping only if it is as close to some target activation as typical target activations are to each other.

Validation via centroid distance. To independently verify that the neighborhoods discovered in the previous step reflect genuine semantic structure rather than noise, we visualize the pairwise centroid cosine distances between all ten concept activation regions. Figure 4 shows the resulting distance matrix, reordered by hierarchical clustering on the centroid distances themselves. Two block structures are immediately visible. The equine cluster (`horse`↔`pony`: 0.158; `horse`↔`donkey`: 0.329; `pony`↔`donkey`: 0.271) exhibit small inter-concept distances, consistent with the high $\hat{\kappa}$ values reported in Figure 1. And semantically isolated concepts such as `castle` have centroid distances exceeding 0.65 to all animal concepts, consistent with castle's low $\hat{\kappa}$.

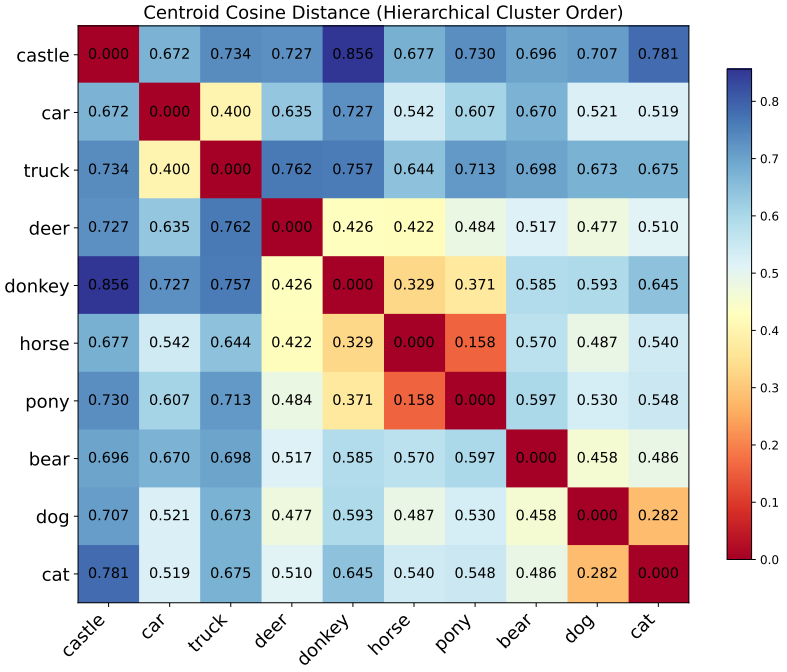


Figure 4: Pairwise centroid cosine distance between concept activation regions at `mid_block.attentions.0`, ordered by hierarchical clustering. Block structure in the matrix corresponds to the neighborhoods discovered by the thresholding procedure described above and to the $\hat{\kappa}$ values reported in Figure 1.

On the use of centroid distance as a proxy. The centroid cosine distance plays a different role from $\hat{\kappa}$ itself, and the two should not be conflated. Definition 4.1 is stated in terms of the volume of fattened activation regions, whose membership is determined by an *infimum* over per-sample distances (the closed ρ-neighborhood $\mathcal{R}_c^{(\rho)}$). The centroid, by contrast, is a single *average* summary of each region.

Centroid distance is used only for *neighborhood discovery* (Definition C.1), an upstream step of deciding which pairs of concepts are close enough to warrant an overlap measurement. We use it here for two reasons. First, it is a stable summary statistic that degrades gracefully under the finite-sample, high-dimensional regime in which we estimate activation regions. Second, an outlier-robust statistic at this stage is conservative: a concept that is filtered out of $\mathcal{N}(c_u)$ contributes zero to the resulting estimator, so under-inclusion only weakens the bound. The estimator $\hat{\kappa}$ is therefore a lower bound on $\kappa(c_u, \rho)$ both because $\mathcal{N}(c_u) \subseteq \mathcal{C} \setminus \{c_u\}$ and because the discovery step itself is conservative. Empirical comparisons against the bound of Theorem 4.6 therefore remain valid.

The $\hat{\kappa}$ statistic itself does not use centroid distance. As specified in the overlap estimation step above, $\hat{\kappa}$ is computed via a nearest-neighbor estimator: for each activation sample $\mathbf{a} \in \mathcal{S}_{c_u}$, we check whether there exists *any* sample $\mathbf{b}$ from *any* concept in $\mathcal{N}(c_u)$ with $d_{\cos}(\mathbf{a}, \mathbf{b}) \leq \rho$. This is the natural finite-sample analogue of the fattened-region membership condition $\mathbf{a} \in \mathcal{R}_{c'}^{(\rho)}$ (with the radius ρ converted to cosine units, see "On the relationship between cosine and ℓ_2 metrics" above), and the fraction of target samples satisfying this condition is a Monte Carlo estimate of the volume

ratio in Definition 4.1 restricted to $\mathcal{N}(c_u)$. The centroid-distance visualization in Figure 4 therefore serves only as an auxiliary consistency check: blocks of small centroid distance should coincide with the discovered $\mathcal{N}(c_u)$, and the high-$\hat{\kappa}$ targets should sit inside such blocks.

Interpretation. Under this estimator, $\hat{\kappa}(c_u) \in [0, 1]$ captures the sample-level coverage of the target region by its neighbors. The results in Figure 1 exhibit a bimodal structure. The four animal-subtype concepts embedded in dense semantic neighborhoods, `dog` ($\hat{\kappa} = 0.89$), `bear` ($\hat{\kappa} = 0.82$), `horse` ($\hat{\kappa} = 0.81$), and `cat` ($\hat{\kappa} = 0.77$), exhibit the highest overlap, consistent with their small inter-centroid distances to fine-grained subtypes (Figure 4). Under our procedure, these concepts each have multiple discovered neighbors, and individual samples from these neighbors fall within the fattening radius ρ of a substantial fraction of target samples. The high overlap is also *robust* to which neighbors are included: in the `horse` pool, several individual neighbors (`mare`, `pony`, `stallion`) each cover 60 to 80% of target samples on their own, and the union across all seven discovered neighbors saturates at $\hat{\kappa} = 0.81$, indicating that overlap is supported by a representational manifold shared across the equine subtype family rather than driven by a single sensitive neighbor choice.

The architectural concept `castle`, by contrast, exhibits substantially lower overlap ($\hat{\kappa} = 0.27$) despite having six discovered neighbors within δ_{c_u}. The distinction is qualitative: castle's neighbors are semantically related but represent *distinct* architectural types rather than fine-grained subtypes of a shared category. They occupy nearby but non-overlapping regions of activation space, so the centroid distances cross the δ_{c_u} threshold while individual sample-level overlap remains limited. Controls like `jellyfish`, `melon`, `police car` are correctly filtered out by the δ_{c_u} step across all five targets and contribute zero to $\hat{\kappa}$, confirming that the estimator is not driven by pool-wide sample density.

This bimodal structure, high-$\hat{\kappa}$ animal subtypes clustered in $[0.77, 0.89]$ and moderate-$\hat{\kappa}$ architecture at 0.27 reflects a property of the model's internal representations at the probed layer: text-to-image diffusion models represent fine-grained subtypes of a common taxonomic category (breeds of dog, equine subtypes) as overlapping regions of a shared activation manifold, while distinct-but-related concepts in other domains maintain more separable representations. Theorem 4.6 predicts that the collateral damage from erasing high-$\hat{\kappa}$ concepts will be substantially greater than from erasing low-$\hat{\kappa}$ concepts; we verify this prediction in Section 5.

$\hat{\kappa}$ measures activation-space overlap, not semantic similarity. A natural concern is whether $\hat{\kappa}$ proxies for some other concept-level property such as semantic category (animal vs. architecture). Our estimator (Appendix C) is computed from sample-level overlap of activation regions, not from category labels or human-judged similarity. `Castle` ($\hat{\kappa} = 0.27$) and the four animal concepts ($\hat{\kappa} \in [0.77, 0.89]$) differ in $\hat{\kappa}$ not because they belong to different taxonomic categories but because castle's discovered neighbors occupy sub-regions of activation space despite being semantically related, while animal subtypes densely overlap their target's region.

C.1 Per-Target Results

We apply the procedure of Appendix C to the five target concepts used throughout Section 5. Each target's pool consists of the target, its ten fine-grained candidate neighbors (equine, feline, canid, ursine, or architectural; Table 5), and three control concepts drawn from {`cactus`, `jellyfish`, `melon`, `police car`}, with `umbrella` in place of `police car` when the target is `car`. Table 2 reports the resulting $\hat{\kappa}$ values together with the size of each discovered neighborhood $|\mathcal{N}(c_u)|$ (Definition C.1), the per-target fattening radius ρ_{c_u}, and the discovered neighbors.

Per-neighbor coverage decomposition. For `horse`, the per-neighbor coverage of target samples (fraction of $\mathcal{S}_{\texttt{horse}}$ with a within-ρ activation in the neighbor's sample set) decomposes as: `mare` (0.78), `pony` (0.72), `stallion` (0.66), `donkey` (0.41), `foal` (0.38), `zebra` (0.30), `mule` (0.22), with the union saturating at $\hat{\kappa} = 0.81$. The strongest individual neighbors (`mare`, `pony`) already cover most of the target region on their own, and several others contribute substantially, indicating that overlap is supported by a representational manifold shared across the equine subtype family rather than being sensitive to which specific neighbor is picked. By contrast, `castle`'s six discovered neighbors each cover only 0.05 to 0.12 of the target region, and their union saturates at $\hat{\kappa} = 0.27$, consistent with castle's neighbors representing distinct architectural types rather than fine-grained subtypes of a shared category.

Target	$\hat{\kappa}$	$\lvert\mathcal{N}(c_u)\rvert$	ρ_{c_u}	Discovered neighbors
dog	0.89	5	0.13	puppy, labrador, husky, beagle, retriever
bear	0.82	5	0.14	grizzly, panda, polar bear, koala, cub
horse	0.81	7	0.15	pony, mare, donkey, stallion, foal, zebra, mule
cat	0.77	5	0.15	kitten, lynx, leopard, tiger, cheetah
castle	0.27	6	0.18	fortress, palace, citadel, keep, manor, cathedral

Table 2: **Per-target $\hat{\kappa}$ estimation results.** For each target c_u, we report the estimated entanglement coefficient $\hat{\kappa}$, the number of discovered neighbors within the centroid-distance threshold δ_{c_u}, the fattening radius ρ_{c_u} (median intra-concept cosine distance), and the discovered neighbor set. castle has a discovered neighborhood of comparable size to the animal targets but $\hat{\kappa}$ roughly $3\times$ lower, isolating semantic density rather than neighborhood cardinality as the driver of $\hat{\kappa}$.

Sensitivity to the neighborhood. For fixed target samples and radius, $\mathcal{N}_1 \subseteq \mathcal{N}_2$ implies $\hat{\kappa}(c;\mathcal{N}_1) \leq \hat{\kappa}(c;\mathcal{N}_2)$, but a new neighbor raises $\hat{\kappa}$ only if it covers target samples not already covered. Adding neighbors to horse in order of centroid distance, mare alone gives $\hat{\kappa} = 0.78$, adding pony gives 0.81, and the remaining five neighbors leave it unchanged. Varying the percentile that defines the discovery threshold from the 10th to the 50th changes $\lvert\mathcal{N}(\texttt{horse})\rvert$ from 3 to 9 and $\lvert\mathcal{N}(\texttt{castle})\rvert$ from 1 to 9, while $\hat{\kappa}$ stays at 0.81 and 0.27. $\hat{\kappa}$ therefore reflects union coverage of the target region rather than neighborhood cardinality.

C.2 Sensitivity of $\hat{\kappa}$ to the choice of probe layer

The $\hat{\kappa}$ estimator fixes the probe layer at $\ell^* =$ mid_block.attentions.0. To assess whether the bimodal $\hat{\kappa}$ structure reported in Figure 1 is an artifact of this choice, we re-ran the estimator at five additional cross-attention layers spanning the U-Net depth, holding all other estimator hyperparameters (target-specific candidate pools, δ_{c_u} percentile, fattening radius rule, prompt templates, timestep $t^* = 25$) fixed.

Layer	castle	cat	horse	bear	dog	r_s vs. mid	Animal std
down_blocks.2.attentions.0	0.34	0.77	0.86	0.92	0.92	+0.97	0.061
down_blocks.2.attentions.1	0.26	0.71	0.74	0.87	0.85	+0.90	0.069
mid_block.attentions.0	**0.27**	**0.77**	**0.81**	**0.82**	**0.89**	—	**0.043**
up_blocks.1.attentions.0	0.09	0.59	0.48	0.88	0.79	+0.80	0.158
up_blocks.1.attentions.1	0.39	0.36	0.61	0.91	0.76	+0.80	0.203
up_blocks.2.attentions.0	0.00	0.78	0.63	0.89	0.88	+0.80	0.105

Table 3: **$\hat{\kappa}$ across six candidate probe layers.** All five alternative layers produce a $\hat{\kappa}$ ranking strongly consistent with mid-block ($r_s \geq 0.80$). Mid-block additionally yields the tightest cluster of animal-subtype $\hat{\kappa}$ values (std $= 0.043$, vs. 0.061 to 0.203 elsewhere), making it the layer at which the "animal-subtype shared-manifold" structure is most coherently expressed.

Conclusion. Three observations support our default choice. First, the $\hat{\kappa}$ *ranking* of the five target concepts is stable across all probed layers ($r_s \geq 0.80$ vs. mid-block in every case): the ordering $\hat{\kappa}(\texttt{castle}) \ll \hat{\kappa}(\{\texttt{cat}, \texttt{horse}, \texttt{bear}, \texttt{dog}\})$ is a property of the underlying concept geometry, not of mid-block specifically. Second, mid-block produces the cleanest bimodal separation between high-entanglement animal subtypes and the low-entanglement architectural concept: the smallest animal $\hat{\kappa}$ exceeds $\hat{\kappa}(\texttt{castle})$ by 0.50 at mid-block, vs. 0.39 to 0.45 at down-block / up-block-1.0 and an inverted ordering at up-block-1.1 where $\hat{\kappa}(\texttt{castle}) = 0.39 > \hat{\kappa}(\texttt{cat}) = 0.36$. Third, mid-block has the lowest within-cluster spread among the four animal concepts (std 0.043, less than half that of every up-block layer), consistent with mid-block being the layer at which fine-grained subtypes are most coherently represented as overlapping regions of a shared manifold. We therefore use mid-block as the default probe for $\hat{\kappa}$ estimation, the visualization in Figure 1, the Lipschitz checks in Appendix E, and the within-target displacement validation in Appendix D.

A note on up_blocks.1.attentions.1**.** The direct displacement measurements in Appendix K are reported at up_blocks.1.attentions.1 rather than mid-block. The $\hat{\kappa}$ ranking at this layer

agrees with mid-block ($r_s = 0.80$, supporting the cross-target $\hat{\kappa}$-scaling reported there), but its bimodal separation is the weakest of the six layers. We use this up-block layer for direct displacement specifically because it lies downstream of the cross-attention edits applied by fine-tuning methods and therefore exhibits the largest absolute displacement signal under STEREO and AdvUnlearn; the κ-scaling claim in Appendix K relies on the Spearman ranking, which is preserved, not on the absolute $\hat{\kappa}$ values at that layer.

Sensitivity to the denoising timestep. Step $t^* = 25$ was chosen a priori as the midpoint of the 50-step DDIM trajectory. Table 4 recomputes $\hat{\kappa}$ at steps 10, 25, and 40 with identical prompts and seeds. Relative to step 25, the concept ranking has $r_s = 0.93$ at step 10 and 0.96 at step 40, with mean neighbor-set Jaccard agreement of 0.98 and 0.82. The ranking is stable, although absolute estimates vary, most notably for `castle` at step 10.

Table 4: $\hat{\kappa}$ at three denoising steps (`mid_block.attentions.0`).

Concept	Step 10	Step 25	Step 40
`castle`	0.77	0.27	0.20
`cat`	0.81	0.77	0.74
`horse`	0.80	0.81	0.84
`bear`	0.81	0.82	0.81
`dog`	0.95	0.89	0.86
`sunglasses`	0.99	0.99	0.99
`eye`	0.99	0.97	0.97

Small and localized concepts. To test whether mean-pooled mid-block activations capture concepts occupying small image regions, we add `sunglasses` and `eye` and probe cross-attention layers at 8×8, 16×16, 32×32, and 64×64 resolution. At step 25, `sunglasses` obtains $\hat{\kappa} = 0.99, 0.88, 0.99, 0.97$ and `eye` obtains 0.97, 0.86, 0.94, 0.94. Discovered neighborhoods remain semantic (`sunglasses`: `eyeglasses`, `goggles`, `shades`, `spectacles`; `eye`: `eyeball`, `eyelid`, `iris`, `pupil`) and no control is selected. The estimator thus detects entanglement for localized concepts, while the exact value of $\hat{\kappa}$ remains probe-dependent.

D Empirical Validation of the Minimum Displacement Assumption

This section provides empirical support for Assumption 4.5 by measuring the activation displacement induced by six unlearning methods and verifying that the displacement propagates to semantic neighbors in proportion to their overlap with the target concept.

Setup. We extract cross-attention activations at `mid_block.attentions.0` at step t^*=25 of 50 DDIM steps from six unlearning methods applied to erase `horse`: ESD, EDiff, SalUn, TRUST, AdvUnlearn, and STEREO. For each method, we compare activations between the original model $\mathcal{D}_\theta$ and the concept-erased model $\mathcal{D}_{\hat{\theta}}$ under matched random seeds, so that observed differences reflect the unlearning edit rather than sampling variance. We measure mean displacement $\|\Phi_{\hat{\theta}}(p) - \Phi_\theta(p)\|_2$ across three prompt categories:

- *Target* prompts containing the erased concept (`horse`);
- *Neighbor* prompts containing semantically related concepts (`pony`, `donkey`, `deer`);
- *Control* prompts containing semantically isolated concepts (`castle`, `car`).

For each prompt category, we use the same 50 shared templates as in Appendix B, differing only in the concept word. Activations are mean-pooled across spatial and token dimensions to produce a single vector per prompt, so the absolute magnitudes reported here are token-averaged and not directly comparable to the unpooled, full-tensor displacement values reported at a different layer in Appendix K (Table 14); the two appendices probe different facets of the propagation mechanism: this one establishes the *ordering* $\Delta_{\text{target}}, \Delta_{\text{neighbor}} > \Delta_{\text{control}}$ within a single target, while Appendix K establishes the $\hat{\kappa}$*-scaling* of Δ_{neighbor} across targets. Figure 5 reports the mean displacement by method and prompt category. Two patterns emerge.

First, all six methods produce measurable target displacement (approximately 6 to 16 in mean-pooled activation L_2 distance), with TRUST producing the largest mean displacement. This pattern is consistent with the minimum-displacement assumption in Assumption 4.5. The magnitudes show how strongly each edit perturbs the probed representation; displacement alone does not measure erasure quality.

TRUST, whose update is gradient-based, produces the largest target displacement (15.82 ± 3.82) and displaces neighbors (8.82 ± 4.88) far more than controls (3.10 ± 1.36), confirming that Assumption 4.5 and the propagation mechanism hold for gradient-based fine-tuning.

Second, the displacement propagates to semantic neighbors: for every method, $\Delta_{\text{neighbor}} \geq \Delta_{\text{control}}$ with the gap ranging from 0.9 (SalUn) to 6.9 (STEREO). This gap is the empirical signature of the propagation mechanism in Theorem 4.6: displacement within $\mathcal{R}_{c_u}$ leaks into overlapping neighbor regions in proportion to κ, causing greater collateral displacement for high-κ neighbors than for low-κ controls. EraseDiff exhibits the smallest cross-category variation, suggesting it operates in a low-selectivity regime where κ-modulated propagation falls below our measurement granularity.

Third, STEREO is the only method for which $\Delta_{\text{neighbor}} > \Delta_{\text{target}}$, while simultaneously achieving the lowest control displacement (2.1). This produces a propagation gap of 6.9, the largest of all six methods. This pattern is consistent with our analysis of STEREO in Remark F.2: by suppressing the entire subspace spanned by the adversarial tokens v_1^*, v_2^* rather than the target alone, STEREO displaces neighbor concepts that lie within this shared subspace by even more than the target itself, while leaving semantically isolated concepts largely intact.

These measurements support two claims central to our analysis. The uniform presence of nontrivial target displacement validates the precondition of Assumption 4.5. The systematic neighbor, control gap, and STEREO's amplified version of it, validate the propagation claim in Step 2 of the proof: methods are not merely displacing the target in isolation but are displacing overlapping regions of activation space, with the magnitude of this collateral displacement modulated by the geometric overlap κ predicted by Theorem 4.6. The robust prediction of the theorem is the *ordering* $\Delta_{\text{neighbor}} > \Delta_{\text{control}}$ within each method and its scaling with κ across concept types, not specific numerical values.

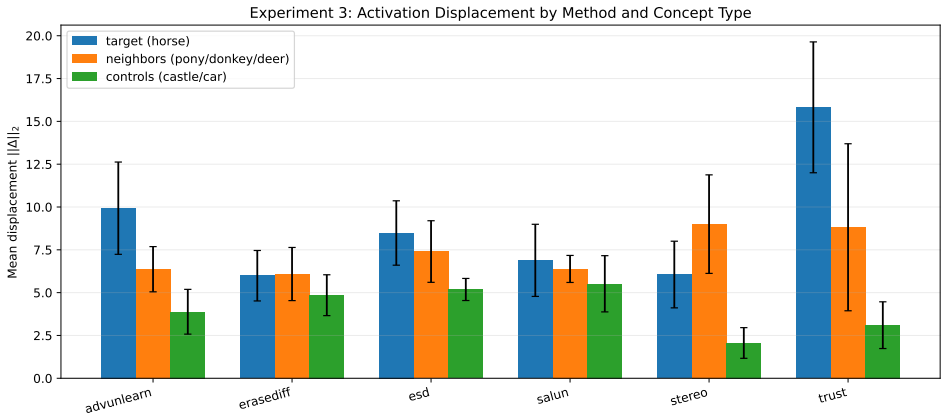


Figure 5: Mean activation displacement $\|\mathbf{\Phi}_{\hat{\theta}}(p) - \mathbf{\Phi}_{\theta}(p)\|_2$ for six unlearning methods erasing `horse`. Target prompts (blue) confirm Assumption 4.5: erasure requires measurable displacement. Neighbor prompts (orange) are displaced more than control prompts (green) for all methods, consistent with the propagation mechanism in Theorem 4.6.

E Empirical Validation of the Lipschitz Assumption

This section provides empirical support for Assumption 4.4 by measuring the relationship between activation distance and change in concept generation probability under natural prompt variations, and verifying that the observed relationship is consistent with a Lipschitz bound with a small empirical constant.

Setup. For each of the ten concepts in Appendix B, we construct a set of prompt pairs (p_b, p_v) where p_b is a base prompt containing the concept and p_v is a variant prompt obtained by one of four perturbations at increasing semantic distance:

- *Synonym substitution:* replacing the concept word with a close synonym (e.g., `horse` $\rightarrow$ `stallion`);
- *Scene change:* altering the background or setting while preserving the concept (e.g., "in a field" $\rightarrow$ "on a beach");
- *Related concept:* substituting a semantically related concept (e.g., `horse` $\rightarrow$ `pony`);
- *Unrelated concept:* substituting an unrelated concept (e.g., `horse` $\rightarrow$ `castle`).

For each pair, we extract cross-attention activations at `mid_block.attentions.0` (step t^*=25) and compute the activation distance $\|\mathbf{\Phi}_\theta(p_v) - \mathbf{\Phi}_\theta(p_b)\|_2$. We then generate images from each prompt and compute the concept generation probability $P(p)$ using a CLIP-based classifier aligned to the base concept. The quantity of interest is the pair $\big(\|\Phi_\theta(p_v) - \Phi_\theta(p_b)\|_2,\ |P(p_v) - P(p_b)|\big)$. This experiment validates the case $\theta' = \theta'' = \theta$ of Assumption 4.4.

Figure 6 plots the concept probability change against the activation distance for all prompt pairs, colored by perturbation type. Two patterns emerge. First, the four perturbation types occupy approximately nested bands of activation distance, with synonyms producing the smallest displacements, scene changes and related-concept substitutions producing intermediate displacements, and unrelated-concept substitutions producing the largest. Second, the change in concept probability scales roughly linearly with activation distance within each band, with no prompt pair exceeding a bound of the form $|P(p_v) - P(p_b)| \leq \hat{L}\,\|\Phi_\theta(p_v) - \Phi_\theta(p_b)\|_2$ for empirical constant $\hat{L} \approx 0.174$.

We emphasize that the empirical constant $\hat{L}$ should be read as a conservative upper bound for this specific layer, timestep, and set of concepts, not as a universal property of Stable Diffusion. Varying the probe layer, the timestep, or the CLIP classifier would likely change the numerical value. The claim we extract from this measurement is weaker and more robust: the concept probability mapping is empirically smooth as a function of activation distance, with no observed violations of the Lipschitz condition across hundreds of prompt pairs spanning four perturbation types. This is the regularity our proof requires. For illustration, combining $\hat{L} \approx 0.174$ with $\varepsilon = 0.05$ yields a minimum displacement bound of $\Delta \geq (1 - \varepsilon)/\hat{L} \approx 5.5$, consistent with the target displacements observed in Appendix D.

F Interpreting Existing Methods Through the Trade-off Lens

The role of κ: why the trade-off severity varies. The overlap coefficient $\kappa(c_u, \rho)$ determines the slope of the Pareto frontier and hence how severe the trade-off is for a given target concept. Concepts embedded in semantically dense neighborhoods (e.g., `horse` among other animals, or `Monet` among Impressionist styles) have large κ and face a steep trade-off. Concepts that are semantically isolated have small κ and can be erased with less collateral damage. This provides actionable guidance: *the difficulty of unlearning a concept depends not only on the method but on the semantic density of its neighborhood.*

Remark F.1 (Tightness of the bound). The bound in Theorem 4.6 is tight when the overlap region $\mathcal{O}$ is exactly a $\kappa(c_u, \rho)$ fraction of $\mathcal{R}_{c_u}^{(\rho)}$ and the displacement field induced by the unlearning edit is uniform over $\mathcal{R}_{c_u}^{(\rho)}$. In practice, displacement is typically non-uniform, since methods may suppress some directions more aggressively than others, in which case the bound is conservative. Characterizing the displacement geometry more precisely is an interesting direction for future work.

Reading existing methods as operating points on the frontier. We now apply this framing to specific unlearning methods, interpreting their empirical behavior as distinct operating points on the predicted Pareto frontier. The discussion is interpretive rather than derivational: we identify where on the frontier each method sits and connect this to the method's mechanism.

Remark F.2 (Why STEREO achieves robustness at the cost of retention). STEREO [Srivatsan et al., 2025] achieves robustness by learning adversarial token embeddings v_1^*, v_2^* that lie near the hardest-to-suppress directions of $\mathcal{R}_{c_u}$, then using a compositional objective to suppress the entire subspace spanned by these directions. This broad suppression corresponds to a large displacement Δ across

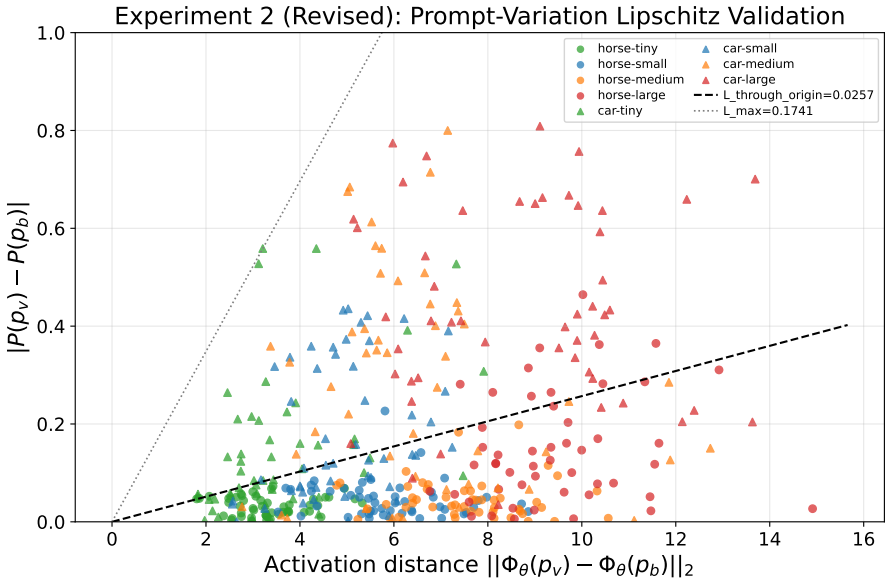


Figure 6: Lipschitz validation via prompt variation. Each point is a (base, variant) prompt pair; the x-axis shows activation distance and the y-axis shows the change in CLIP-based concept probability. Points are colored by semantic distance from the base prompt (green: synonyms; blue: scene changes; orange: related concepts; red: unrelated concepts). All points lie below the empirical bound $\hat{L} = 0.174$ (dotted line), supporting Assumption 4.4.

$\mathcal{R}_{c_u}$, which by Theorem 4.6 directly implies large collateral displacement in $\mathcal{R}_{c'}$ for all neighbors c' with $\kappa(c_u, \rho) > 0$. Our empirical observation that `donkey` and `zebra` generation degrades after `horse` erasure with STEREO (Section 3) is consistent with this prediction.

Remark F.3 (Why SAeUron preserves neighbors but remains vulnerable). SAeUron [Cywiński and Deja, 2025] achieves unlearning by identifying a small number of sparse autoencoder features (τ_c features per concept) that are highly specific to the target concept, then ablating only those features during inference. Because the intervention is restricted to a sparse set of concept-specific features rather than applied across the full activation region $\mathcal{R}_{c_u}$, the effective displacement Δ is small and concentrated in a low-dimensional subspace. This limits collateral damage to neighbors (small γ), explaining SAeUron's superior retention performance in sequential unlearning. However, the sparsity of the intervention also means that portions of $\mathcal{R}_{c_u}$ remain unsuppressed, precisely the portions reachable through indirect correlated prompts, consistent with the concept leakage observed in Section 3. In the language of our framework, SAeUron achieves small γ at the cost of large ε.

Remark F.4 (Intervention breadth determines the operating point). More generally, a method's operating point on the (ε, γ) frontier is set by how broadly it displaces activations across $\mathcal{R}_{c_u}$, not by whether it edits weights or intervenes at inference time. Methods that displace activations broadly, whether by fine-tuning (SalUn, AdvUnlearn, STEREO) or at inference time (SLD), achieve small ε at the cost of larger γ. Methods whose intervention is narrow, such as SEOT and SAeUron, achieve small γ but leave larger ε because the intervention does not cover the full activation region. This is a direct consequence of Theorem 4.6: the effective displacement Δ is the lever, and methods differ in how broadly they apply it.

G Proof of Theorem 4.6

We restate the theorem and provide the proof. Throughout, distances in $\mathcal{H}$ are measured in ℓ_2 norm.

Proof. We proceed in three steps.

Step 1: Required activation displacement for ε-robustness.

By construction of the activation region $\mathcal{R}_{c_u}$ (Definition A.1), every $\mathbf{h} \in \mathcal{R}_{c_u}$ corresponds to a prompt p whose pre-edit generation contains c_u, i.e. $g_{c_u}(\theta, p) = 1$. After an ε-robust edit (Definition 4.2), $g_{c_u}(\hat{\theta}, p) \leq \varepsilon$. Applying Assumption 4.4 to the pre- and post-edit activations (with $\theta' = \theta$, $\theta'' = \hat{\theta}$, $p' = p$) gives, for every p with $\mathbf{\Phi}_\theta(p) \in \mathcal{R}_{c_u}$,

$$1 - \varepsilon \;\leq\; \left| g_{c_u}(\theta, p) - g_{c_u}(\hat{\theta}, p) \right| \;\leq\; L \, \|\mathbf{\Phi}_{\hat{\theta}}(p) - \mathbf{\Phi}_\theta(p)\|_2. \tag{18}$$

Combined with Assumption 4.5, this gives a uniform displacement floor on $\mathcal{R}_{c_u}$:

$$\Delta \;\geq\; \frac{1 - \varepsilon}{L}. \tag{19}$$

Step 2: Aggregate overlap induces aggregate collateral displacement.

By Definition 4.1,

$$\kappa(c_u, \rho) = \frac{\mathrm{vol}\Big(\mathcal{R}_{c_u}^{(\rho)} \cap \bigcup_{c' \in \mathcal{C} \setminus \{c_u\}} \mathcal{R}_{c'}^{(\rho)}\Big)}{\mathrm{vol}\Big(\mathcal{R}_{c_u}^{(\rho)}\Big)}. \tag{20}$$

Let $\mathcal{O} := \mathcal{R}_{c_u}^{(\rho)} \cap \bigcup_{c' \neq c_u} \mathcal{R}_{c'}^{(\rho)}$ denote the overlap region. By Definition 4.3, the non-target prompt distribution $\mathcal{P}_{-u} = \frac{1}{|\mathcal{C} \setminus \{c_u\}|} \sum_{c' \neq c_u} \mathcal{P}_{c'}$ is supported on the set of activations that lie in $\bigcup_{c' \neq c_u} \mathcal{R}_{c'}^{(\rho)}$. We invoke the following regularity assumption, which we make explicit:

(Mass-volume regularity). The push-forward measure of $\mathcal{P}_{-u}$ under $\mathbf{\Phi}_\theta$ places mass on $\mathcal{O}$ in proportion to its volume share of the support, i.e. $\Pr_{p \sim \mathcal{P}_{-u}}[\mathbf{\Phi}_\theta(p) \in \mathcal{O}] \geq \kappa(c_u, \rho)$. This holds when activations within each $\mathcal{R}_{c'}^{(\rho)}$ are sampled approximately uniformly under $\mathcal{P}_{c'}$.

For any prompt $p \sim \mathcal{P}_{-u}$ with $\mathbf{\Phi}_\theta(p) \in \mathcal{O}$, we have $\mathbf{\Phi}_\theta(p) \in \mathcal{R}_{c_u}^{(\rho)}$, so the displacement floor from Step 1 applies (extending Assumption 4.5 from $\mathcal{R}_{c_u}$ to its closed ρ-neighborhood by continuity of the displacement field): $\|\mathbf{\Phi}_{\hat{\theta}}(p) - \mathbf{\Phi}_\theta(p)\|_2 \geq \Delta$. Combining with the mass-volume regularity gives

$$\begin{aligned} \mathbb{E}_{p \sim \mathcal{P}_{-u}}\left[\|\mathbf{\Phi}_{\hat{\theta}}(p) - \mathbf{\Phi}_\theta(p)\|_2\right] \;&\geq\; \Pr_{p \sim \mathcal{P}_{-u}}[\mathbf{\Phi}_\theta(p) \in \mathcal{O}] \cdot \Delta \\ &\geq\; \kappa(c_u, \rho) \cdot \Delta. \end{aligned} \tag{21}$$

Step 3: Deriving the trade-off inequality.

Substituting the displacement floor (19) into (21) gives

$$\mathbb{E}_{p \sim \mathcal{P}_{-u}}\left[\|\mathbf{\Phi}_{\hat{\theta}}(p) - \mathbf{\Phi}_\theta(p)\|_2\right] \geq \kappa(c_u, \rho) \frac{1 - \varepsilon}{L}. \tag{22}$$

Since Definition 4.3 bounds the left-hand side above by γ, we obtain

$$\gamma \geq \kappa(c_u, \rho) \frac{1 - \varepsilon}{L}. \tag{23}$$

Rearranging (for $\kappa(c_u, \rho) > 0$) yields

$$\varepsilon \geq 1 - \frac{L\gamma}{\kappa(c_u, \rho)}. \tag{24}$$

This completes the proof. □

Remark on metric compatibility. The proof uses the ℓ_2 metric throughout because Assumption 4.4 is stated in ℓ_2. Our $\hat{\kappa}$ estimator (Appendix C) instead uses cosine distance to characterize region membership. These two choices are compatible at the level of the geometric objects in the bound: for unit-normalized activations, $\|\mathbf{a} - \mathbf{b}\|_2^2 = 2\, d_{\cos}(\mathbf{a}, \mathbf{b})$, so the two metrics induce the same topology on the activation sphere and define the same fattened regions $\mathcal{R}_c^{(\rho)}$ and entanglement coefficient $\kappa(c_u, \rho)$ up to a monotone re-parameterization of the radius ρ. This re-parameterization is absorbed into the choice of ρ in our empirical $\hat{\kappa}$ estimator. Consequently, estimating κ via cosine distance while applying the theoretical bound under an ℓ_2 Lipschitz assumption is internally consistent: the bound's set-theoretic content (volume overlap of ρ-neighborhoods) transfers between metrics, while the Lipschitz constant L remains tied to the ℓ_2 formulation in which Assumption 4.4 is stated.

H Implementation Details

Here we introduce the implementation of each unlearning method used in Section 5, including hyperparameters, training protocols, and compute budgets. All methods are applied to CompVis Stable Diffusion v1.4 [Rombach et al., 2022]. The full implementation, including code, prompt CSVs, and evaluation scripts, is available at our public repository: `https://github.com/Jazzssmine/concept-entangle`

H.1 Method-Specific Hyperparameters

We group the 13 benchmarked methods into three mechanism families. *Closed-form editing methods* (UCE, RECE) modify model weights through analytic updates. *Fine-tuning methods* (ESD, SalUn, EDiff, CPE, TRUST, AdvUnlearn, STEREO) modify model weights through gradient-based optimization. *Inference-time methods* (SEOT, SAeUron, SLD, SAFREE) intervene during generation without permanently modifying the base model. AdvUnlearn and STEREO additionally incorporate adversarial training, explicitly targeting the aggressive-erasure end of the trade-off; their inclusion tests whether robustness gains incur measurable cost to neighbor preservation.

Unless otherwise noted, we use the hyperparameters specified in each method's original publication. We document deviations and choices not fully specified by the original papers in the per-method descriptions below.

UCE [Gandikota et al., 2024]. Unified Concept Editing uses closed-form updates to text-to-image attention projections to erase target concepts without gradient training. Configuration: guide concept **unconditional**, erase scale **1.0**, preserve scale **1.0**, regularization λ **0.5**, no explicit preserve concepts; each concept is erased independently.

RECE [Gong et al., 2024]. Reliable and Efficient Concept Erasure iteratively discovers embeddings that recover the target concept and removes their effect through closed-form cross-attention edits. Configuration: initialized from base SD v1.4, edit epochs **1**, erase scale **1.0**, preserve scale **0.1**, λ **0.1**, adversarial-embedding regularization **0.1**, guide concept **unconditional**.

ESD [Gandikota et al., 2023]. Erased Stable Diffusion fine-tunes the cross-attention layers to suppress the target concept via a negative-guidance objective. We use the ESD-sd (stable-diffusion) variant. Configuration: learning rate **5e-5**, training steps **200**, negative guidance scale $\eta = \mathbf{5}$, batch size **1**. Training uses the target concept name as the single forget prompt. For the cross-architecture experiments (Appendix M), we adapt ESD to each backbone. On SDXL (base-1.0) we use the ESD-x variant, which fine-tunes only the U-Net cross-attention layers (`attn2` query, key, value, and output projections; about 546M parameters). Configuration: Adam, learning rate **2e-4**, training steps **200**, batch size **1**, negative guidance scale $\eta = \mathbf{1}$. On FLUX.1-schnell we use a stricter ESD-x variant that fine-tunes only the attention key and value projections (`to_k`, `to_v`) in all 19 joint and 38 single transformer blocks (about 1.08B parameters), keeping the text-side projections (`add_k_proj`, `add_v_proj`) frozen. Configuration: learning rate **1e-5**, training steps **500**, negative guidance scale $\eta = \mathbf{2}$.

SalUn [Fan et al., 2023]. Saliency-guided Unlearning identifies salient weights via gradient magnitudes and applies random labeling on the forget set to those weights only. We use saliency threshold sparsity **0.5**, forget-set size **800**, learning rate **1e-5**, and training epochs **5**. The forget and retain sets consist of 800 target and 800 non-target images generated by Stable Diffusion v1.4.

EDiff [Wu et al., 2024]. EraseDiff fine-tunes the UNet using a bi-level optimization that explicitly balances erasure and retention objectives. Configuration: learning rate **1e-5**, inner steps per outer update **2**, total outer iterations **300**.

CPE [Lee et al., 2025] Concept Pinpoint Eraser trains nonlinear residual attention gates with an anchoring loss to suppress target concepts while preserving other concepts. Configuration: **5** stages of **450** iterations each (stage 1 uses $4\times$ the iterations and $10\times$ the learning rate), learning rate **3e-5**, gate rank **16**, erasure guidance **0.3**, PAL weight **1e5**, noise **1e-3**; adversarial learning with learning rate **1e-2**, **450** iterations, and **16** added prompts per stage.

TRUST [Kori et al., 2026] TRUST dynamically identifies target-associated neurons and selectively fine-tunes them with Hessian-based regularization. Configuration: concept neurons are heads whose $|z\text{-score}| > \mathbf{2.0}$ on the CLIP-loss attribution; learning rate **1e-4**, batch size **2**, **2** epochs of **60** steps (final checkpoint), **25** sampling steps for neuron localization.

AdvUnlearn [Zhang et al., 2024]. Adversarial Unlearning augments the erasure objective with adversarial prompt generation to improve robustness against indirect concept recovery. We use adversarial prompt budget 30 per training step, adversarial learning rate **1e-3**, main-model learning rate **1e-5**, and training steps **1000**. The attack prompts are updated per training steps.

STEREO [Srivatsan et al., 2025]. STEREO proceeds in two phases: a Search for Trainable Embeddings via RObust optimization (STE) phase that learns adversarial token embeddings v_1^*, v_2^*, followed by a Robust Erasure via Orthogonalization (REO) phase that suppresses the subspace spanned by these embeddings. STE phase: **200** steps, learning rate **5e-6**. REO phase: **200** steps, learning rate **2e-5**, $K = \mathbf{2}$ adversarial embeddings per target.

SEOT [Li et al., 2024]. Soft Embedding Optimization for Target-concept erasure modifies the text embedding at inference time to suppress the target concept without altering model weights. We use embedding regularization strength **0.01** and optimization steps per generation **10**.

SAeUron [Cywiński and Deja, 2025]. SAeUron uses sparse autoencoder feature ablation applied at inference: for each target, τ_c concept-specific SAE features are identified from activation statistics, and these features are zeroed during generation. We use the pre-trained SAE released by the authors, with $\tau_c = \mathbf{1}$ features ablated per target.

SLD [Schramowski et al., 2023]. Safe Latent Diffusion adds safety guidance during denoising to suppress target content without retraining the model. Configuration: the **SLD-Max** preset, i.e. safety guidance scale **5000**, warm-up steps **0**, threshold **1.0**, momentum scale **0.5**, momentum β **0.7**; **50** sampling steps, CFG scale **7.5**.

SAFREE [Yoon et al., 2025]. SAFREE filters prompt embeddings away from a target-concept subspace and adapts its intervention during denoising without updating model weights. Configuration: concept subspace from **12** per-concept negative terms (synonyms, sub-breeds, and phrasings such as "a photo of a dog"), α **-0.5**, re-attention up to step **20**, FreeU hyperparameters **(1.0, 1.0, 0.9, 0.2)**; **50** sampling steps, CFG scale **7.5**.

H.2 Compute and Infrastructure

All experiments were run on GPU A100 40GB. Unlearning per method-target pair requires approximately 1 GPU-hours for fine-tuning methods and 2 GPU-hours for inference-time methods. Evaluation (image generation + classification) requires approximately 0.5 GPU-hours per method-target pair.

H.3 Reproducibility

For each (method, target) pair, we generate 100 images from direct prompts, 100 to 235 from indirect prompts (depending on how many valid indirect prompts the target yields; 235 for `castle`, 200 for `dog`, 195 for `horse`, and 100 for every other target), 50 per neighbor candidate (500 total), and 50 per control concept (150 total); IRA is computed over the 650 neighbor and control images. All images use fixed random seeds, with seed equal to the row index in the target's prompt CSV. This ensures seed-level variation is controlled when comparing methods on the same prompt. The base model $\mathcal{D}_\theta$ is loaded from the HuggingFace checkpoint `CompVis/stable-diffusion-v1-4`. Code, prompt CSVs, and the classification vocabulary (Table 5) will be released to enable full reproduction of Table 1.

Use of existing assets. Our experiments use publicly released assets from prior work: the CompVis Stable Diffusion v1.4 checkpoint, released under the CreativeML OpenRAIL-M license; LAION metadata/captions, released under CC-BY 4.0 while the underlying images remain subject to their

original copyrights; and the released SAeUron sparse-autoencoder features, released under Apache-2.0. We cite the original sources for these assets and use them only for research evaluation under their stated release terms and licenses.

I Evaluation Setup and Metric Definitions

This appendix provides the full evaluation protocol summarized in Section 5.1: the prompt construction pipeline for direct, indirect, neighbor, and control evaluation; formal definitions of the six metrics used in Section 5; and the classification vocabulary shared across all experiments.

I.1 Prompt Construction and Scoring Pipeline

Our automated evaluation protocol is designed to measure two coupled consequences of concept entanglement: *indirect recovery*, where an erased concept remains recoverable through correlated contextual cues, and *collateral forgetting*, where suppressing a target concept degrades performance on nearby concepts that should remain intact. To distinguish these effects, we separately construct (i) direct prompts that explicitly name c, (ii) indirect prompts constructed from correlated context without naming c, (iii) neighbor prompts that explicitly name each concept in the neighborhood $\mathcal{N}(c)$, and (iv) control prompts that name each concept in the control set $\mathcal{C}(c)$.

Correlated context extraction. For each target concept c, we collect a target-matched caption subset from LAION [Schuhmann et al., 2022] aligned with diffusion-model pretraining. Specifically, we identify captions that explicitly contain c or its lexical variants, and extract co-occurring content words and short phrases from this subset. We remove stopwords, punctuation, low-information function words, and lexical forms that directly mention the target concept. The remaining candidates are ranked using corpus-level association statistics and filtered to obtain a final set of correlated context words that captures the contextual footprint of c in the training distribution. For example, when $c =$ `horse`, the resulting set includes terms such as `jockey`, `stable`, `saddle`, `racetrack`, and `polo`, for $c =$ `castle`, it includes `moat`, `turret`, `medieval`, `knight`, and `drawbridge`.

To visualize how these correlated prompts organize in text-embedding space, we pool prompts that explicitly mention the target concept and prompts that omit the target token but retain correlated context, then project their CLIP text embeddings to two dimensions with PCA. Figure 7 shows that context-rich prompts occupy a nearby region rather than a disjoint cluster, supporting our bag-of-words motivation that concept evidence is distributed across correlated tokens rather than isolated in a single concept word.

Indirect recovery prompts. We use the correlated context set for c to construct a set of indirect prompts $\mathcal{P}_{\mathrm{ind}}(c)$. These prompts are generated under an explicit lexical exclusion constraint: they must not contain the target token or excluded lexical variants of c. Intuitively, $\mathcal{P}_{\mathrm{ind}}(c)$ probes whether the erased concept can still be elicited through contextual associations alone, without direct lexical mention. To improve prompt diversity while preserving control, we generate multiple prompts from different subsets of the correlated context words and retain only prompts that satisfy the exclusion rule.

Semantic neighbors and control concepts. The semantic neighborhood $\mathcal{N}(c)$ is determined by the procedure of Definition C.1: $\mathcal{N}(c)$ collects the candidates in c's own pool whose centroid cosine distance to c falls below the 25th percentile δ_c of pairwise distances in that pool. This produces concept-specific neighborhoods of five to seven discovered neighbors per target (Table 2). The control set $\mathcal{C}(c)$ of each target consists of three control concepts drawn from {`cactus`, `jellyfish`, `melon`, `police car`}, with `umbrella` in place of `police car` when the target is `car`; these are concepts whose centroids lie well above δ_c for every target, verified to contribute zero to $\hat{\kappa}$ across all five targets. For each neighbor $c' \in \mathcal{N}(c)$ and control $\tilde{c} \in \mathcal{C}(c)$, we generate prompts that explicitly mention the concept while excluding lexical forms of the target c. Per-target prompt files are released in the supplementary material.

Classifier-based scoring. All images are classified using a CLIP-based classifier over a fixed vocabulary of 65 concept labels (Appendix I.3). For each image, the classifier returns the top-1 label

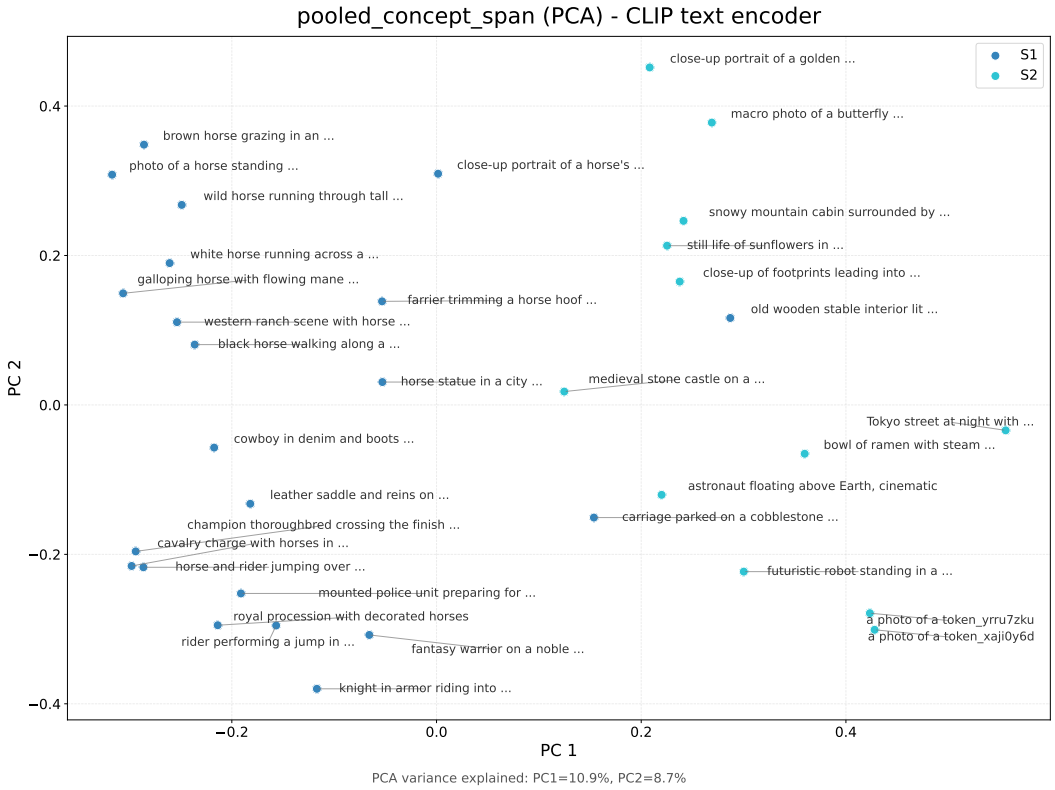


Figure 7: **Context-only prompts remain close to target prompts in CLIP text-embedding space.** PCA projection of pooled prompts used for the `horse` case study. One group explicitly contains the concept token, while the other omits it but retains correlated context words (e.g., `jockey`, `saddle`, `racetrack`). The proximity between these groups illustrates why token-level erasure can remain vulnerable to indirect contextual prompts.

among the vocabulary. All metrics defined below are computed from these top-1 assignments. In several random instances, we also perform a manual evaluation to ensure consistency of the results.

I.2 Evaluation Metrics

Each metric is defined with respect to a base model $\mathcal{D}_\theta$ and an unlearned model $\mathcal{D}_{\hat{\theta}}$. We write $\mathrm{Acc}(\mathcal{D}, c, \mathcal{P})$ for the fraction of images generated by model $\mathcal{D}$ from prompts in set $\mathcal{P}$ that are classified as concept c by the CLIP evaluator.

Erasure metrics (measuring ε). These metrics measure how effectively the target concept c is suppressed in the unlearned model.

- *Unlearning Accuracy (UA)*: The fraction of images generated from direct prompts $\mathcal{P}_{\mathrm{dir}}(c)$ by $\mathcal{D}_{\hat{\theta}}$ that are *not* classified as c.

$$\mathrm{UA}(c) = 1 - \mathrm{Acc}(\mathcal{D}_{\hat{\theta}}, c, \mathcal{P}_{\mathrm{dir}}(c)). \tag{25}$$

  Higher UA indicates more complete erasure under explicit lexical prompts.

- *Indirect Recovery Rate (IRR)*: The fraction of images generated from indirect recovery prompts $\mathcal{P}_{\mathrm{ind}}(c)$ by $\mathcal{D}_{\hat{\theta}}$ that *are* classified as c:

$$\mathrm{IRR}(c) = \mathrm{Acc}(\mathcal{D}_{\hat{\theta}}, c, \mathcal{P}_{\mathrm{ind}}(c)). \tag{26}$$

  Higher IRR indicates greater vulnerability to concept recovery through correlated contextual cues. In contrast to UA, IRR measures the ε-robust erasure property formalized in Definition 4.2.

Retention metrics (measuring γ). We measure retention at three levels of granularity:

- *In-domain Retain Accuracy (IRA)*: The fraction of images whose intended concept is not the target, i.e., all images from neighbor and control prompts, that are classified as the concept they were meant to show. Direct and indirect recovery prompts are excluded because their intended label is the target itself. IRA therefore measures how well the unlearned model still renders every non-target concept it is asked for, without normalizing by the base model.
- *Neighbor Preservation (NP)*: This is our primary trade-off metric. The overlap with remaining concepts is often dominated by the overlap with a few concepts which are mostly semantically similar with the target concept. Therefore, as a proxy metric for utility preservation, we evaluate utility preservation in the semantic neighborhood $\mathcal{N}(c)$ of the target concept (Definition C.1) and define

$$\mathrm{NP}(c) = \frac{1}{|\mathcal{N}(c)|} \sum_{c' \in \mathcal{N}(c)} \frac{\mathrm{Acc}(\mathcal{D}_{\hat{\theta}}, c', \mathcal{P}_{\mathrm{dir}}(c'))}{\mathrm{Acc}(\mathcal{D}_{\theta}, c', \mathcal{P}_{\mathrm{dir}}(c'))}. \tag{27}$$

  An NP score of 1.0 indicates perfect preservation of neighbors after unlearning; lower values indicate proportional collateral damage. The ratio form normalizes out the base-model recognition accuracy, which differs across concepts (and which is generally below 1.0 under a 65-class vocabulary).
- *Control preservation (CP)*: To distinguish neighbor-specific collateral damage from broader degradation, we additionally evaluate performance on a non-neighbor control set $\mathcal{C}(c)$. We define

$$\mathrm{CP}(c) = \frac{1}{|\mathcal{C}(c)|} \sum_{\tilde{c} \in \mathcal{C}(c)} \frac{\mathrm{Acc}(\mathcal{D}_{\hat{\theta}}, \tilde{c}, \mathcal{P}_{\mathrm{dir}}(\tilde{c}))}{\mathrm{Acc}(\mathcal{D}_{\theta}, \tilde{c}, \mathcal{P}_{\mathrm{dir}}(\tilde{c}))}. \tag{28}$$

  Comparing NP and CP isolates neighbor-specific collateral damage from broader degradation: a method with $\mathrm{NP} \ll \mathrm{CP}$ is damaging concepts selectively in the target's neighborhood, consistent with the propagation mechanism in Theorem 4.6. A method with $\mathrm{NP} \approx \mathrm{CP} \ll 1$ is causing uniform degradation across the concept space.

DamageGap. A summary statistic we introduce to quantify neighbor-selective damage: $\mathrm{DamageGap}(c) = \mathrm{CP}(c) - \mathrm{NP}(c)$. A positive DamageGap indicates that neighbors are damaged more than controls, consistent with the trade-off predicted by Theorem 4.6. A DamageGap near zero indicates uniform degradation (or uniform preservation) across both sets.

I.3 Classification Vocabulary

All CLIP-based classification uses the following 65-concept vocabulary, which is the union of the five targets, their candidate neighbors, and ten out-of-domain control labels, which include the controls used for $\hat{\kappa}$ estimation (Appendix C). The same vocabulary is used for every metric and every one of these five targets, ensuring cross-target comparability. For the five additional targets of Appendix J.6, the vocabulary is extended with each new target and its candidate neighbors (Table 12).

I.4 Classifier reliability

We evaluate the fixed CLIP classifier on 100 labeled images per target (500 in total), drawn from the base model's generations, and report per-concept true-positive and false-positive rates (Table 6). Scoring is one-vs-rest among the five targets: for each target, its 100 images are the positives and the 400 images of the other four targets are the negatives, so TPR is out of 100 and FPR is out of 400. The classifier attains a micro-averaged TPR of 97.0% and FPR of 0.55%, ranging from 90% TPR for `dog` to 1.25% FPR for `horse`. All methods share the same classifier, vocabulary, prompts, and decision rule, so classifier error does not differ across methods.

J Per-Concept Results

This appendix provides the per-concept evaluation tables underlying the averaged results in Table 1. For each of the five target concepts, we report all metrics across the seven original unlearning methods.

Category	Concepts
Targets	horse, cat, dog, bear, castle
Equine	pony, donkey, zebra, mustang, mare, foal, stallion, mule, bison, camel
Feline	kitten, tiger, lion, leopard, panther, cheetah, lynx, jaguar, cougar, bobcat
Canid	puppy, fox, coyote, jackal, dingo, hyena, husky, labrador, beagle, retriever
Ursine	grizzly, polar bear, panda, cub, koala, black bear, brown bear, teddy bear, sloth bear, sun bear
Architectural	fortress, tower, cathedral, palace, keep, citadel, watchtower, stronghold, fortified wall, manor
Controls	airplane, bicycle, cactus, jellyfish, lighthouse, melon, police car, teapot, violin, volcano

Table 5: The 65-concept classification vocabulary used for all CLIP-based metric evaluations in Section 5. The vocabulary is the union of the five targets, their respective fine-grained subtype candidates spanning four animal category and one architectural category, and ten out-of-domain controls. See Table 2 for the per-target subset $\mathcal{N}(c)$ that survives the δ_c-filter and forms the averaging set for NP.

Table 6: CLIP classifier reliability on labeled images.

Target	TPR (%) ↑	FPR (%) ↓
dog	90	0.25
cat	100	0.25
bear	96	1.00
horse	99	1.25
castle	100	0.00
Average	97.0	0.55

The concepts span estimated entanglement coefficients $\hat{\kappa} \in [0.27, 0.89]$ (Figure 1), enabling the κ-scaling analysis in Section 5.3.

We order tables by $\hat{\kappa}$ in descending order to make the predicted scaling visually apparent: collateral damage (low IRA, low NP, high DamageGap) is most pronounced for high-$\hat{\kappa}$ concepts and attenuates as $\hat{\kappa}$ decreases.

SAeUron coverage. SAeUron's released sparse autoencoder features cover the four animal concepts (dog, bear, horse, cat) but not castle. We mark the corresponding row with dashes in Table 11 and exclude SAeUron from the averaged castle entries. Attempts to substitute SAeUron's architectures feature for castle did not yield meaningful suppression of castle generation (verified by visual inspection at the default ablation multiplier); we therefore report SAeUron only on concepts where its released features apply.

Reading the tables. All tables follow the same column structure as Table 1: erasure metrics (UA, IRR), retention metrics (IRA, NP, CP), and the composite DamageGap. The *Original* row reports baseline rates for the unedited $\mathcal{D}_\theta$ on each concept's evaluation set; note that baseline IRA varies substantially across concepts (e.g., 0.85 for cat vs. 0.54 for tower/castle) due to differences in CLIP evaluator accuracy on each concept's in-domain object set. This per-concept baseline variation motivates our use of normalized retention ratios in Section 5.3.

J.1 Dog

Observations. Dog exhibits the most severe trade-off in our evaluation, consistent with its position as the highest-$\hat{\kappa}$ concept. STEREO achieves complete erasure (UA = 100) but collapses NP to 1.81 and IRA to 4.76, effectively destroying the model's ability to generate dog-related neighbors. SAeUron shows analogous catastrophic damage (NP = 4.27, IRA = 8.76) despite operating at inference time, suggesting that even modest displacement can wreck retention when neighbor activation regions densely overlap the target. ESD stands out as the only method achieving meaningful

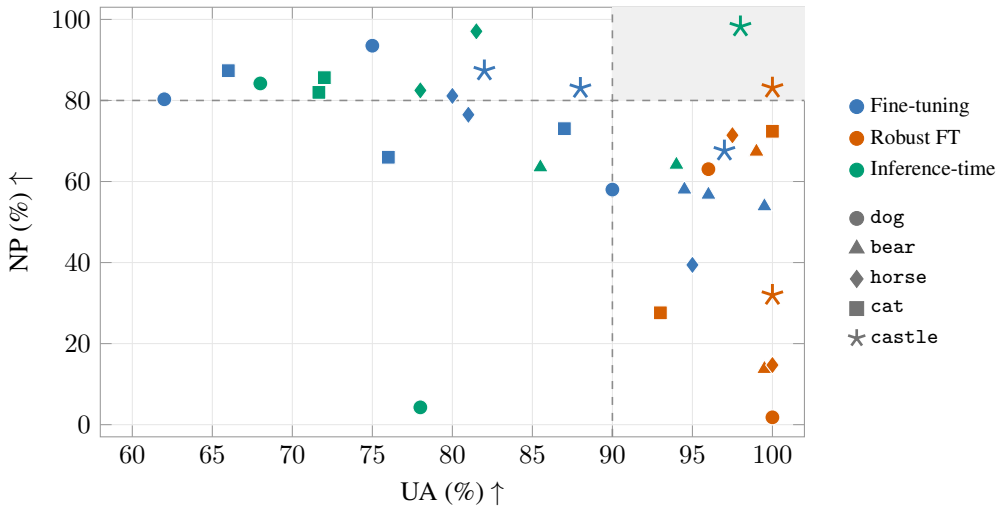


Figure 8: **Per-concept operating points of the seven original methods.** Each point is one (method, concept) pair from Appendix J; color encodes method family and marker encodes the target concept. The shaded region marks strong erasure together with strong neighbor preservation (UA > 90, NP > 80). No point for the four high-$\hat{\kappa}$ animal targets enters it; the only points that do are `castle` ($\hat{\kappa} = 0.27$) under SEOT and AdvUnlearn, consistent with the κ-scaled bound of Theorem 4.6. SAeUron has no `castle` point (Appendix J.5).

		Erasure		Retention			
Method	**Type**	UA↑	IRR↓	IRA↑	NP↑	CP↑	DamageGap↓
Original	–	0	27.0	78.92	100	100	–
ESD	FT	75.0	17.5	74.15	93.51	96.0	2.48
SalUn	FT	90.0	8.0	50.92	58.01	76.0	17.98
EDiff	FT	62.0	23.5	66.61	80.29	92.99	12.70
AdvUnlearn	Robust FT	96.0	20.0	74.30	63.05	99.0	35.94
STEREO	Robust FT	100.0	1.0	4.76	1.81	64.0	62.18
SEOT	Inf.	68.0	30.0	79.07	84.20	99.0	14.80
SAeUron	Inf.	78.0	18.5	8.76	4.27	73.0	68.72

Table 7: **Per-concept evaluation for `dog` ($\hat{\kappa} = 0.89$).** Discovered neighbors include `puppy`, `labrador`, `beagle`, `husky`, and `retriever` (full list in Appendix I.1). As the highest-$\hat{\kappa}$ concept in our evaluation, `dog` exhibits the most severe robustness-retention trade-off: robust fine-tuning methods (STEREO, AdvUnlearn) achieve high UA but cause substantial IRA and NP degradation.

erasure (UA = 75) while preserving neighbors well (NP = 93.51, DamageGap = 2.48), at the cost of higher residual leakage (IRR = 17.5). The Pareto separation between ESD $(75, 93.51)$ and STEREO $(100, 1.81)$ is a ≈ 92-point NP gap for a 25-point UA gain, the steepest trade-off observed across our concepts and a direct empirical instantiation of the κ-scaled lower bound predicted by Theorem 4.6.

J.2 Bear

Observations. `Bear` sits in the high-$\hat{\kappa}$ regime alongside horse and dog, and exhibits the predicted Pareto structure. STEREO again occupies the high-erasure / low-retention extreme (UA = 99.5, NP = 13.75, DamageGap = 63.25), while AdvUnlearn achieves comparable erasure (UA = 99) with substantially better neighbor preservation (NP = 67.37). The clustering pattern is unusually compressed on bear: standard fine-tuning methods (ESD, SalUn, EDiff) and the inference-time methods cluster tightly between NP = 53.89 to 67.37, with SEOT and AdvUnlearn essentially tied at NP ≈ 64 to 67. Notably, no method on bear matches the (high UA, high NP) operating point achieved on lower-$\hat{\kappa}$ concepts: even ESD, the best performer on `dog` and `castle`, reaches only NP

		Erasure		Retention			
Method	**Type**	UA↑	IRR↓	IRA↑	NP↑	CP↑	DamageGap↓
Original	–	0	85.5	62.75	100	100	–
ESD	FT	94.5	9.0	44.00	58.02	95.0	36.97
SalUn	FT	99.5	5.5	51.51	53.89	89.0	35.10
EDiff	FT	96.0	6.0	38.61	56.75	94.0	37.24
AdvUnlearn	Robust FT	99.0	1.0	42.46	67.37	100	32.62
STEREO	Robust FT	99.5	2.0	15.88	13.75	77.0	63.25
SEOT	Inf.	94.0	13.5	50.84	64.14	96.48	32.34
SAeUron	Inf.	85.5	15.0	22.37	63.50	93.8	30.30

Table 8: **Per-concept evaluation for** `bear` **($\hat{\kappa} = 0.82$).** Discovered neighbors include `grizzly`, `panda`, `polar bear`, `koala`, and `cub` (full list in Appendix I.1). Image counts differ for two runs: SalUn uses 272 neighbor images and SEOT uses 462 neighbor images.

= 58.02 here. This compression of the achievable retention region is a direct manifestation of the $\hat{\kappa}$-scaled Pareto frontier predicted by Theorem 4.6.

J.3 Horse

		Erasure		Retention			
Method	**Type**	UA↑	IRR↓	IRA↑	NP↑	CP↑	DamageGap↓
Original	–	0	75.5	71.23	100	100	–
ESD	FT	80.0	33.5	62.92	81.12	97.0	15.87
SalUn	FT	95.0	28.5	40.92	39.42	91.99	52.57
EDiff	FT	81.0	31.0	63.84	76.48	94.0	17.51
AdvUnlearn	Robust FT	97.5	23.0	65.38	71.42	100	28.57
STEREO	Robust FT	100.0	7.5	20.74	14.71	27.0	12.28
SEOT	Inf.	81.5	38.5	69.69	97.06	99.0	1.93
SAeUron	Inf.	78.0	39.5	76.92	82.48	94.0	11.51

Table 9: **Per-concept evaluation for** `horse` **($\hat{\kappa} = 0.81$).** Discovered neighbors include `pony`, `mare`, `donkey`, `zebra`, `stallion`, `foal` and `mule` (full list in Appendix I.1). Image counts differ for two runs: *Original* uses approximately 480 neighbor and control images, and SEOT uses 58 direct images.

Observations. `Horse` is the running example used throughout Section 5 and exhibits the canonical Pareto structure across method families. STEREO occupies the high-erasure / catastrophic-retention extreme (UA = 100, NP = 14.71, IRA = 20.74); SEOT and SAeUron occupy the high-retention / weaker-erasure region (NP $\geq$ 82 but IRR $\geq$ 38.5); standard fine-tuning methods sit in between, with ESD reaching a clean $(80, 81.12)$ operating point at DamageGap = 15.87. SalUn is an outlier among standard FT methods, achieving high UA = 95 with NP = 39.42, far below ESD and EDiff at comparable erasure, consistent with SalUn's gradient-saliency masking displacing more activation than necessary for erasure. A second notable observation is STEREO's collapsed CP = 27, far below STEREO's CP on the other four concepts (range 64 to 96): in this high-$\hat{\kappa}$ regime, STEREO's adversarial subspace appears to spill beyond the intended object subspace into stylistically-correlated features, manifesting the broad subspace suppression discussed in Remark F.2.

J.4 Cat

Observations. `Cat` exhibits a trade-off structure consistent with the high-$\hat{\kappa}$ regime, though its base IRA = 85.50 is the highest of the five concepts, providing more retention "headroom" than other targets. STEREO again occupies the high-erasure / catastrophic-retention extreme (UA = 93, NP = 27.60, IRA = 27.68), with DamageGap = 62.40 matching its damage on horse and bear. AdvUnlearn achieves complete erasure (UA = 100, IRR = 0) at NP = 72.40, demonstrating that adversarial

		Erasure		Retention			
Method	**Type**	UA↑	IRR↓	IRA↑	NP↑	CP↑	DamageGap↓
Original	–	0	96.0	85.50	100	100	–
ESD	FT	66.0	38.0	66.92	87.35	98.0	10.65
SalUn	FT	87.0	21.0	34.53	73.04	100	26.95
EDiff	FT	76.0	24.0	62.30	65.99	100	34.00
AdvUnlearn	Robust FT	100.0	0.0	50.76	72.40	98.0	25.60
STEREO	Robust FT	93.0	7.0	27.68	27.60	90.0	62.40
SEOT	Inf.	71.66	34.0	42.15	81.99	100	18.01
SAeUron	Inf.	72.0	37.5	60.25	85.63	96.0	10.37

Table 10: **Per-concept evaluation for `cat` ($\hat{\kappa} = 0.77$).** Discovered neighbors include `kitten`, `lynx`, `leopard`, `tiger`, and `cheetah` (full list in Appendix I.1). The original baseline IRA = 85.50 reflects high CLIP-evaluator accuracy on cat's in-domain object set after correcting for kitten/cat label confusion (see Appendix I). Image counts differ for two runs: *Original* uses approximately 480 neighbor and control images, and SEOT uses 72 direct images.

training can preserve neighbors more effectively than STEREO at comparable erasure strength. ESD preserves neighbors well (NP $= 87.35$, DamageGap $= 10.65$) but achieves the weakest erasure on this concept (UA $= 66$, IRR $= 38$). The STEREO IRA-retention ratio for cat ($27.68/85.50 = 0.32$) is consistent with the corresponding ratio on horse (0.29) and bear (0.25), reinforcing the $\hat{\kappa}$-scaling result in Section 5.3.

J.5 Castle

		Erasure		Retention			
Method	**Type**	UA↑	IRR↓	IRA↑	NP↑	CP↑	DamageGap↓
Original	–	0	9.5	53.84	100	100	–
ESD	FT	88.0	6.5	48.15	83.03	99.0	15.98
SalUn	FT	97.0	1.0	54.76	67.58	94.0	26.41
EDiff	FT	82.0	2.5	46.46	87.35	99.0	11.65
AdvUnlearn	Robust FT	100.0	1.0	44.00	83.10	100	16.89
STEREO	Robust FT	100.0	0.0	29.38	31.96	96.0	64.03
SEOT	Inf.	98.0	6.5	53.84	98.21	99.0	0.79
SAeUron	Inf.	–	–	–	–	–	–

Table 11: **Per-concept evaluation for `castle` ($\hat{\kappa} = 0.27$).** Discovered neighbors include `fortress`, `palace`, `citadel`, `keep`, `cathedral` and `manor` (full list in Appendix I.1). SAeUron's released sparse autoencoder features do not cover `castle`; we attempted to substitute its `architectures` feature, but the resulting ablation did not yield meaningful suppression of castle generation. Image counts differ for two runs: *Original* uses approximately 480 neighbor and control images, and SEOT uses 60 direct and 100 indirect images.

Observations. `Castle` is the lowest-$\hat{\kappa}$ concept in our evaluation and demonstrates qualitatively different trade-off behavior from the high-$\hat{\kappa}$ animal concepts. Several methods achieve strong erasure with minimal retention damage: ESD reaches UA = 88 at NP = 83.03 (DamageGap = 15.98), AdvUnlearn achieves complete erasure (UA = 100) at NP = 83.10 (DamageGap = 16.89), and SEOT reaches UA = 98 at NP = 98.21 (DamageGap = 0.79, the lowest of any (method, concept) pair in our evaluation). This is qualitatively distinct from the high-$\hat{\kappa}$ concepts, where no method achieves both UA ≥ 90 and NP ≥ 80. STEREO is the notable exception: despite the low predicted κ-floor, STEREO incurs NP = 31.96 and DamageGap = 64.03, comparable to its damage on the high-$\hat{\kappa}$ animal concepts. This is the empirical illustration of Theorem 4.6 as a *lower bound* rather than a ceiling: STEREO's REO objective broadly suppresses the spanned adversarial subspace regardless of whether neighbors actually occupy the displaced region (Remark F.2), producing damage well above

the κ-predicted floor that other methods successfully hug. The separation between STEREO and the alternatives on `castle` suggests room for κ-aware unlearning methods that achieve robust erasure without STEREO's κ-blind over-suppression.

J.6 Additional Target Concepts

To extend the five-target analysis in the main text, we evaluate `snake`, `fish`, `tree`, `butterfly`, and `car`. Their estimated entanglement coefficients span $\hat{\kappa} \in [0.37, 0.83]$, adding concepts between the low-overlap `castle` and the densely overlapping animal subtypes. Table 13 reports the unedited reference and results for EDiff, ESD, SalUn, and STEREO using the metrics defined in Section 5. Averaged over the four methods, NP is 65.0, 62.0, 61.7, 61.3, and 58.8 for `snake`, `fish`, `tree`, `butterfly`, and `car`, decreasing monotonically with $\hat{\kappa}$ ($r_s = -1.0$). Within-method correlations are -0.4 (EDiff), -0.7 (ESD), -0.2 (SalUn), and -0.7 (STEREO). STEREO again sits at the aggressive end, reaching UA $\geq$ 99.1 on every new target with NP between 16.0 and 26.6. Table 12 lists the neighbors discovered for each new target with the procedure of Appendix C. The classification vocabulary for these targets is extended accordingly (Appendix I.3).

Target	$\hat{\kappa}$	$\lvert\mathcal{N}(c_u)\rvert$	**Discovered neighbors**
`snake`	0.37	7	anaconda, boa, cobra, garter snake, king snake, python, rattlesnake
`fish`	0.52	8	bass, cod, eel, herring, mackerel, salmon, trout, tuna
`tree`	0.63	6	cedar, cypress, maple, oak, pine, spruce
`butterfly`	0.66	7	birdwing, dragonfly, fritillary, monarch butterfly, moth, red admiral, swallowtail
`car`	0.83	7	convertible, coupe, hatchback, minivan, sedan, suv, van

Table 12: **Discovered neighbors for the five additional targets.** As for the original targets (Table 2), neighborhood size does not determine $\hat{\kappa}$: `fish` has the most neighbors but an intermediate $\hat{\kappa}$.

Table 13: Results for the five additional target concepts. Original denotes the pretrained model before unlearning. All evaluation metrics are percentages. Higher UA, IRA, NP, and CP are better; lower IRR and DamageGap are better. DamageGap is the difference between control preservation (CP) and neighbor preservation (NP).

Concept	Method	$\hat{\kappa}$	UA (%) ↑	IRR (%) ↓	IRA (%) ↑	NP (%) ↑	CP (%) ↑	DamageGap (%) ↓
Snake	Original	0.37	9.2	6.0	31.1	100.0	100.0	0.0
	EraseDiff	0.37	88.1	8.6	19.1	81.2	93.6	12.4
	ESD	0.37	97.9	6.8	20.6	81.3	98.1	16.8
	SalUn	0.37	100.0	0.0	5.5	72.4	86.2	13.8
	STEREO	0.37	100.0	0.0	14.8	25.1	96.0	71.1
Fish	Original	0.52	6.2	11.0	40.5	100.0	100.0	0.0
	EraseDiff	0.52	92.3	5.3	23.2	78.0	92.1	14.1
	ESD	0.52	94.5	2.7	34.5	79.2	98.0	18.8
	SalUn	0.52	100.0	0.0	14.9	67.5	97.8	30.3
	STEREO	0.52	100.0	0.0	5.8	23.2	38.4	15.2
Tree	Original	0.63	4.4	61.8	51.8	100.0	100.0	0.0
	EraseDiff	0.63	32.0	55.2	53.8	75.1	98.0	22.9
	ESD	0.63	39.7	48.0	50.3	82.4	98.9	16.5
	SalUn	0.63	100.0	0.0	5.8	62.7	88.0	25.3
	STEREO	0.63	100.0	0.0	4.9	26.6	32.0	5.4
Butterfly	Original	0.66	7.1	12.2	36.0	100.0	100.0	0.0
	EraseDiff	0.66	46.2	37.3	34.6	74.7	92.5	17.8
	ESD	0.66	72.6	28.9	31.2	78.1	83.3	5.2
	SalUn	0.66	100.0	0.0	0.9	73.1	93.9	20.8
	STEREO	0.66	100.0	0.0	10.5	19.2	68.0	48.8
Car	Original	0.83	2.6	60.0	87.5	100.0	100.0	0.0
	EraseDiff	0.83	47.4	49.3	69.2	78.7	91.2	12.4
	ESD	0.83	60.5	28.6	75.8	73.8	98.6	24.8
	SalUn	0.83	79.3	16.1	17.8	66.8	85.6	18.8
	STEREO	0.83	99.1	4.3	21.1	16.0	75.2	59.1

J.7 Which targets admit clean erasure

Across the per-concept results we report (the seven original methods on the five main targets, Tables 7 to 11, and four methods on the five additional targets, Table 13), the only (method, concept) pairs that reach both UA > 90 and NP > 80 are on the two lowest-$\hat{\kappa}$ targets: `castle` ($\hat{\kappa} = 0.27$; SEOT at $98.0/98.21$, AdvUnlearn at $100.0/83.10$) and `snake` ($\hat{\kappa} = 0.37$; ESD at $97.9/81.3$). No method reaches this region on any target with $\hat{\kappa} \geq 0.52$ (Figure 8 and Table 13). Because $\hat{\kappa}$ is computed from base-model activations alone, it identifies these targets before any unlearning is run, which is the sense in which we propose it as a pre-deployment diagnostic (Section 6). The damage is also neighbor-selective rather than global: control preservation stays above 85 for twelve of the thirteen methods in Table 1 (all except STEREO), while NP drops.

K Direct Validation: Activation Displacement Scales with $\hat{\kappa}$

Theorem 4.6 predicts that activation displacement at neighbor prompts scales linearly with κ for a method achieving ε-robust erasure. The Neighbor Preservation (NP) measure used in Section 5 is a probability-units proxy for this displacement, related through Assumption 4.4. To test Theorem 4.6's prediction in the units the theorem is stated in, we measure activation displacement directly:

$$\Delta_{\text{neighbor}}(c_u) \;=\; \mathbb{E}_{p_{c'} \sim \mathcal{N}(c_u)} \big[\, \big\| \Phi_{\hat{\theta}}(p_{c'}) - \Phi_{\theta}(p_{c'}) \big\|_2 \,\big]$$

We measure Δ_{neighbor} for each fine-tuning method on each of the five target concepts using 200 neighbor prompts per concept, with activations captured from `up_blocks.1.attentions.1` at DDIM step 25 of 50. Activations are flattened across spatial and token dimensions before computing L_2 distances, so absolute magnitudes here are larger than the mean-pooled token-level magnitudes reported in Appendix D (Figure 5); both views are consistent in ordering and in the prediction they probe (within-target ordering in Appendix D, cross-target $\hat{\kappa}$-scaling here). Inference-time methods are omitted: SEOT's text-embedding intervention does not propagate measurably to this layer in L_2 distance, and SAeUron's sparse-feature ablation produces displacement values an order of magnitude larger than fine-tuning methods, reflecting a real mechanism difference rather than direct comparability with weight-update methods.

Displacement scales monotonically with $\hat{\kappa}$. Table 14 reports mean neighbor displacement across the five concepts and five fine-tuning methods. All five methods exhibit a positive Spearman rank correlation between $\hat{\kappa}$ and displacement, four of them at $r_s \geq 0.70$, with ESD showing strict monotonicity ($r_s = +1.00$) and STEREO closely tracking ($r_s = +0.90$ with a single inversion between `cat` and `horse`, whose $\hat{\kappa}$ values differ by only 0.04). Across all five methods, displacement on `castle` ($\hat{\kappa} = 0.27$) is substantially lower than the average displacement across the four high-$\hat{\kappa}$ animal concepts: ratios range from $1.68\times$ (ESD) to $3.18\times$ (STEREO). The qualitative form predicted by Theorem 4.6, displacement increasing with $\hat{\kappa}$, is empirically supported.

Method	`castle` ($\hat{\kappa}$=0.27)	`cat` (0.77)	`horse` (0.81)	`bear` (0.82)	`dog` (0.89)	r_s vs. $\hat{\kappa}$
ESD	128.4	194.8	213.6	221.0	235.4	$+1.00$
STEREO	75.9	234.2	233.4	240.9	257.0	$+0.90$
AdvUnlearn	71.6	110.5	150.9	123.2	123.7	$+0.70$
EDiff	67.8	163.0	178.1	172.1	175.6	$+0.70$
SalUn	88.1	220.7	215.6	220.4	215.3	$+0.10$

Table 14: **Mean activation displacement at neighbor prompts.** Each cell reports Δ_{neighbor} in L_2 activation distance, averaged over 200 neighbor prompts. r_s is computed across the five concepts. Note that $n = 5$ concepts limits the statistical power of significance testing on individual methods; we report r_s as a descriptive monotonicity measure. The consistency of positive correlations across all five methods (range $+0.10$ to $+1.00$; four of five at ≥ 0.70) provides aggregate evidence for the κ-scaling predicted by Theorem 4.6.

STEREO's displacement is consistent with a single local Lipschitz constant. Theorem 4.6 predicts that for a method operating exactly at the bound, the ratio $\kappa(1-\varepsilon)/(\Delta_{\text{neighbor}} + c_0)$ should equal a single Lipschitz constant L across all concepts. We compute this ratio for each method using

STEREO's saturated $\varepsilon \approx 0.035$ (mean IRR across the five concepts; see Table 1) and an additive constant $c_0 = 0.05$. The resulting L_{tight} values are reported in Table 15. STEREO's L_{tight} varies by only 7.8% across the five concepts ($L \in [0.0032, 0.0035]$); other methods exhibit spreads of 24% to 57%. STEREO's single-L consistency is precisely the quantitative prediction of Theorem 4.6: a method operating at the bound should exhibit a κ-and-concept-independent local Lipschitz constant. Among the five methods evaluated, only STEREO satisfies this prediction, consistent with STEREO occupying the high-Δ, low-ε regime where the bound is tightest.

Method	`castle`	`cat`	`horse`	`bear`	`dog`	Spread
STEREO	0.0035	0.0032	0.0034	0.0033	0.0034	**7.8%**
EDiff	0.0039	0.0046	0.0045	0.0047	0.0050	23.6%
SalUn	0.0030	0.0034	0.0037	0.0036	0.0040	29.5%
ESD	0.0021	0.0039	0.0037	0.0036	0.0037	53.4%
AdvUnlearn	0.0037	0.0068	0.0053	0.0065	0.0070	57.1%

Table 15: **Per-concept L_{tight} values.** $L_{\text{tight}} = \kappa(1-\varepsilon)/(\Delta_{\text{neighbor}} + c_0)$ is the Lipschitz constant value that places Theorem 4.6's bound exactly at the observed displacement. If a method operates at the bound, L_{tight} should be approximately constant across concepts. STEREO is the only method exhibiting this consistency (7.8% spread), supporting its operation at the bound's saturated-erasure regime.

Local versus global Lipschitz. The local $L \approx 0.0034$ inferred from STEREO's displacement is much smaller than the global $\hat{L} = 0.174$ reported in Figure 6. This reflects the well-known phenomenon that probability changes saturate near classification decision boundaries: the global Lipschitz constant measured across diverse prompt-pair perturbations averages over both saturated and unsaturated regimes, while the local constant near $\mathcal{R}_{c_u}$ during erasure reflects unsaturated behavior specifically. The two values are compatible measurements of different but related quantities; principled estimation of L that distinguishes these regimes is a direction for future work.

L Text-Space versus Denoiser Entanglement

Our analysis measures entanglement inside the denoising network. To examine whether overlap is already present in the text encoder, we apply the same estimator, candidate pools, and prompts to text-encoder representations.

T5 (FLUX.1-schnell). We extract padding-masked, mean-pooled token representations from T5 blocks 0, 11, and 23 and from the final encoder output for the five targets. At all four layers and for all five targets, $\hat{\kappa} = 0$. This is not caused by unrelated concepts entering the neighborhoods: no control is selected at any layer, 19 of 20 target-layer conditions have positive silhouette scores, and the nearest final-layer centroids are semantically appropriate (`dog`/`puppy`, `cat`/`kitten`, `horse`/`pony`, `castle`/`palace`), yet lie 1.61 to $2.97\times$ beyond the discovery threshold. The same pools and estimator give $\hat{\kappa} = 0.27, 0.77, 0.81, 0.82, 0.89$ in the SD v1.4 denoiser.

CLIP text space versus denoiser within SDXL. To compare the two spaces within one pipeline, we estimate $\hat{\kappa}$ for a ten-concept set spanning several semantic groups in SDXL's CLIP text-embedding space and in its denoiser cross-attention space (Table 16). In CLIP space, nine of ten concepts obtain $\hat{\kappa} = 1.0$ with semantically related neighborhoods; only `castle` obtains 0. The denoiser produces heterogeneous values, including 0.72 for `horse`, 0.68 for `car`, and 0.60 for `dog`, with the remaining concepts at 0. Neighbors here are discovered only within this ten-concept set, the visualization pool of Appendix B, rather than from each target's full candidate pool, so these values are not comparable to $\hat{\kappa}_{\text{SDXL}}$ in Table 17; they serve only to contrast the two representation spaces under identical conditions.

Interpretation. Text-space geometry varies across encoders: CLIP shows substantial overlap under our estimator, whereas T5 shows none. Neither comparison is a controlled variance decomposition (pooling differs between text and denoiser features, and the T5 comparison spans model families), so we read them as evidence that the source and severity of entanglement depend on both the text

Table 16: $\hat{\kappa}$ for the ten-concept set in SDXL's CLIP text-embedding space and in its denoiser cross-attention space.

Concept	$\hat{\kappa}$ (CLIP text)	$\hat{\kappa}$ (denoiser)
`horse`	1.00	0.72
`pony`	1.00	0.00
`donkey`	1.00	0.00
`deer`	1.00	0.00
`dog`	1.00	0.60
`cat`	1.00	0.00
`bear`	1.00	0.00
`car`	1.00	0.68
`truck`	1.00	0.00
`castle`	0.00	0.00

representation and the denoiser computation, not as proof that one encoder is less entangled in general.

M Cross-Architecture Experiments

SDXL. SDXL shares the cross-attention structure of SD v1.4, so $\hat{\kappa}$ is directly computable. Table 17 reports ESD under the same erasure and evaluation protocol for three concepts. Figure 9 shows that SDXL's cross-attention activations have qualitatively similar geometry to SD v1.4 (Figure 1): related animal concepts occupy overlapping regions, and `castle` lies apart from them. Castle's own architectural neighbors are not in this visualization, and its estimated entanglement with them is higher on SDXL ($\hat{\kappa}_{\text{SDXL}} = 0.72$) than on SD v1.4 (0.27). The absolute level of entanglement therefore shifts across architectures, but $\hat{\kappa}$ and observed collateral damage agree within each: on SDXL, NP orders `cat` > `castle` > `dog` exactly as $\hat{\kappa}_{\text{SDXL}}$ predicts (Table 17).

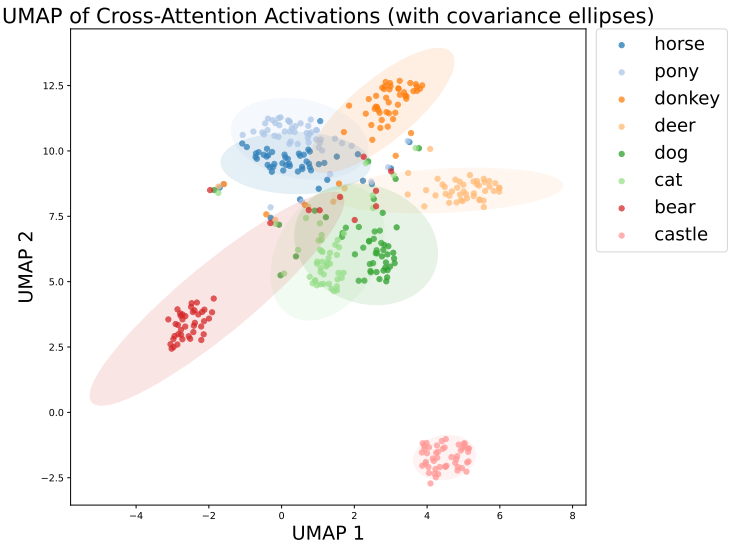


Figure 9: UMAP projection of SDXL cross-attention activations for eight concepts, with covariance ellipses. As in SD v1.4 (Figure 1), related animal concepts (`horse`/`pony`/`donkey`, `dog`/`cat`) overlap, while `castle` lies apart from the animal clusters. Castle's architectural neighbors are not shown; its entanglement with them is $\hat{\kappa}_{\text{SDXL}} = 0.72$ (Table 17).

FLUX (MMDiT with T5). FLUX uses joint text-image attention and has no cross-attention block analogous to `mid_block.attentions.0`. A proxy κ estimated from the joint-attention denoiser saturates near 1 for all targets (`dog` 1.00, `cat` 0.97, `castle` 0.97) and also for two unrelated controls (`airplane`→`teapot` 0.99, bootstrap 95% CI $[0.94, 1.00]$; `teapot`→`airplane` 1.00, CI

Table 17: ESD on SDXL. NP decreases monotonically with $\hat{\kappa}_{\text{SDXL}}$.

Concept	$\hat{\kappa}_{\text{SDXL}}$	UA ↑	NP ↑	DamageGap ↓
`cat`	0.65	99	90.9	9.1
`castle`	0.72	91	85.0	15.0
`dog`	0.92	77	77.1	22.9

$[0.95, 1.00]$; centroid distance 0.011). The proxy therefore does not resolve concept structure at the layers we can access, and we treat κ-scaling on MMDiT as untested. Table 18 nonetheless shows that erasure produces neighbor-selective damage: NP collapses while unrelated controls are preserved.

Table 18: ESD on FLUX (percent).

Concept	UA ↑	IRR ↓	NP ↑	CP ↑	DamageGap ↓
Base (`dog`)	8.2	26.5	100.0	100.0	0.0
`dog`	78.0	8.5	35.6	96.9	61.3
`cat`	83.0	9.5	37.8	95.3	57.6
`castle`	85.0	20.0	40.5	98.9	58.3

Comparison across backbones. For ESD on `dog`, erasure is nearly matched across backbones (UA 75, 77, and 78 on SD v1.4, SDXL, and FLUX), while NP is 93.5, 77.1, and 35.6. The robustness-retention trade-off therefore persists on newer backbones rather than being specific to SD v1.4. Absolute NP is not strictly comparable across architectures because their image distributions differ.

N Broader Impacts

Our analysis is instantiated in diffusion models, but the mechanism only requires activation displacement, local smoothness of the activation map, and target-neighbor overlap in representation space. Similar κ-scaled trade-offs may arise in autoregressive language models, classifiers, and multimodal models, but verifying this requires task-specific definitions of concepts, activations, and preservation.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and Introduction state the paper's main claims: a geometric account of concept entanglement, a method-agnostic impossibility/trade-off result, and empirical validation across thirteen unlearning methods on SD v1.4, with additional targets and SDXL/FLUX experiments.

   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: Section 7 discusses limitations: our targets are object concepts only (styles, identities, and harmful content are untested); cross-architecture evidence covers SDXL with three concepts and one method, and κ-scaling on MMDiT models such as FLUX remains untested; and $\hat{\kappa}$ is an empirical proxy whose absolute value depends on the probe layer, denoising timestep, and candidate pool. It also notes uncertainty about generalization beyond diffusion models.

   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 main theoretical result is Theorem 4.6, with Definitions 4.1 to 4.3, and Assumption 4.4 to 4.5 stated before the theorem. The main text gives a proof sketch, Corollary 4.8 states the impossibility consequence, and Appendix G provides the full proof.

   Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: The paper provides the base model, target concepts, unlearning methods, evaluation metrics, prompt construction, activation-extraction protocol, and per-method hyperparameters in Section 5.1 and Appendices H and I. Appendix H also states that code and evaluation scripts are available through a public repository.

   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: Appendix H provides a public repository for the implementation, prompts, and evaluation scripts, and Appendices H and I describe the training/evaluation setup needed to reproduce the main tables and figures. The released materials are intended to cover the main experiments and baselines.

   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 5.1 specifies the base model, generation settings, target concepts, unlearning methods, and metrics. Appendix H gives method-specific hyperparameters and compute details, while Appendix I describes prompt construction, scoring, and the classification vocabulary.

   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: Table 1 reports mean $\pm$ sample standard deviation across target concepts for the main averaged experimental results. The caption specifies that variability is computed across concepts.

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 H.2 reports the hardware and approximate compute requirements: experiments were run on A100 40GB GPUs, with approximate GPU-hour costs for fine-tuning, inference-time methods, and evaluation per method-target pair.

   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 studies concept unlearning and evaluates existing methods using object-category prompts and generated images. The authors have reviewed the NeurIPS Code of Ethics and are not aware of any aspect of the research that violates it.

   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: The paper discusses broader impact in Appendix N.

    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 new high-risk pretrained model, image generator, or scraped dataset. It uses existing models and datasets and releases code, prompts, and evaluation scripts rather than a deployable generative model requiring additional safeguards.

    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: We use publicly released assets including Stable Diffusion v1.4, LAION-derived captions, and released sparse-autoencoder features from SAeUron, and cite the corresponding sources in the main text and appendices. Appendix H states that these assets are used for research evaluation under their stated release terms and licenses.

    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: [Yes]

    Justification: The paper introduces new evaluation assets such as prompts, classification vocabulary, and evaluation scripts, and Appendix H states that these are released through a public repository. Appendix I documents prompt construction, scoring, and the vocabulary used for evaluation.

    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 paper does not involve crowdsourcing experiments 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 paper does not involve crowdsourcing or research with human subjects, 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 methodology does not use LLMs as an important, original, or non-standard component. The work studies text-to-image diffusion models and concept unlearning; any LLM use limited to writing, editing, or formatting would not require declaration under this question.

    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.