Unlearning without Forgetting: Securely Removing Targeted Concepts from Large-Scale Vision-Language Open-Vocabulary Detectors

Supplementary Material

7. Related Work

Open-vocabulary object detection (OvOD). OvOD aligns visual features with text to localize arbitrary categories beyond fixed label sets, leveraging pre-trained vision-language models (VLMs) like CLIP [41] and SigLIP [64]. Detectors such as GLIP [29], GroundingDINO [35], DetCLIP series [62], and MM-Grounding DINO [67] learn scalable region-word alignment, while LLMDet [23] enriches representations using LLM-generated captions. OvOD and broader VLM techniques have also expanded into diverse multimodal and risk-sensitive domains [8, 24, 26–28, 46, 55, 56, 66]. However, these models rely on web-scale data that often includes privacy-sensitive or non-consensual content. Since retraining from scratch to remove specific concepts is prohibitively expensive, selective machine unlearning for OvOD [57] is increasingly critical.

Machine unlearning in deep models. Practical erasure is typically achieved via optimization (e.g., Gradient Ascent [34, 48], Negative Preference Optimization [7, 65]), distillation to match reference models on retained data [69], or output-space smoothing like LOTUS [45]. For multimodal tasks, methods like MultiDelete [9] and cross-modal decoupling [5, 6] separate embeddings to enable selective erasure. While effective for LLMs and closed-set vision, directly balancing forget and retain losses at the output level fails in OvOD. It ignores the decomposable structure of VLM embeddings, inducing geometric entanglement interference where forgetting updates leak into shared semantic directions, thereby degrading retention and zero-shot generalization.

Image and VLM unlearning methods. Beyond LLMs, image unlearning techniques leverage saliency-based weight updates (SalUn [13]), gradient rectification (GDR-GMA [31]), and Nash bargaining (MUNBa [53]). For generative models, EraseDiff [54] manipulates score functions, though this is structurally incompatible with discriminative detection. In VLMs, CLIPErase [60] suppresses target alignment in the joint embedding space, and CAGUL [2] utilizes cross-modal attention guidance. While these VLM methods outperform image-domain approaches on OvOD, none resolve the core issue of geometric entanglement in decomposable vision-language embeddings. This critical gap directly motivates our proposed null-space projection approach.

Unlearning benchmarks and unified testing. Standardized benchmarks are crucial for evaluating unlearning efficacy and utility. Recent frameworks like TOFU [38], MUSE [44], and OpenUnlearning [11] provide rigorous evaluation metrics for LLMs. MLLMU-Bench [37] extends this to multimodal models using fictitious profiles for QA tasks. However, these existing benchmarks predominantly focus on text generation and multimodal QA. They do not address the unique challenges of region-level localization, strong cross-modal coupling, and the non-detectable requirements inherent to open-vocabulary detection. This significant gap highlights the necessity for an OvOD-specific unlearning benchmark like our UOD-Bench.

8. Extended Details of UOD-Bench and Experimental Setup

8.1. UOD-Bench Construction Details

This section provides detailed construction steps for UOD-Bench. The benchmark construction follows a four-step pipeline:

Step 1: Vocabulary unification. We map category names from COCO [32] and V3Det [52], along with region-level phrases from GoldG [29], into a unified phrase space. All concepts are then stratified into four frequency tiers (Ultra-High, High, Medium, Low) based on their occurrence statistics.

Step 2: Forget/retain concept sampling. For each target forget ratio $r_f \in \{0.01, 0.05, 0.10, 0.15\}$, we select a subset of concepts as the *forget* set following a hierarchical vocabulary sampling scheme that balances across frequency tiers. The remaining concepts form the *retain* set. Importantly, the sampled forget/retain concept lists for each ratio are fixed and shared across all three tasks (OD, PG, REC), ensuring that all tasks measure unlearning on the same semantic concepts.

Step 3: Task-specific benchmark instantiation. Given the fixed forget/retain vocabulary, we construct three task-specific benchmarks with different annotation formats and evaluation protocols:

- **OD.** We construct COCO-based forget/retain splits by filtering Object Detection and Visual Grounding (ODVG)-style region annotations according to the benchmark vocabulary, and evaluate whether the detector suppresses forgotten concepts while preserving detection performance on retain concepts.
- **PG.** Based on GoldG (GQA+Flickr30k), we create forget/retain splits over region–phrase pairs, where the concept for splitting is defined by the (normalized) head noun of each phrase.

Table 7. Comprehensive statistics of OvOD Unlearning Benchmarks. Vocab denotes the number of unique phrases in the complete set.

Benchmark	Ratio	Forget Set		Retain Set		Complete Set		Vocab
		Images	Regions	Images	Regions	Images	Regions	
OD	1%	50	289	4,851	35,092	4,901	35,381	2,541
OD	5%	250	1,898	4,662	34,018	4,912	35,916	2,525
OD	10%	499	3,168	4,405	32,576	4,904	35,744	2,483
OD	15%	750	4,993	4,162	30,806	4,912	35,799	2,537
PG	1%	39	85	4,853	26,843	4,892	26,928	4,564
PG	5%	187	263	4,880	26,298	5,067	26,561	4,425
PG	10%	411	726	4,844	26,986	5,255	27,712	4,672
PG	15%	643	1,273	4,875	27,024	5,518	28,297	4,739
REC	1%	50	50	4,950	4,950	5,000	5,000	1,253
REC	5%	250	250	4,950	4,950	5,200	5,200	1,372
REC	10%	500	500	4,950	4,950	5,450	5,450	1,371
REC	15%	724	724	4,950	4,950	5,674	5,674	1,406

- **REC.** Starting from the same GoldG data, we construct referring expression examples by identifying the primary visual referent in each caption (e.g., "fresh banana", "woman with umbrella") from the region-level phrase annotations. For each sample, we retain only the bounding box corresponding to this referent. Forget/retain splits are determined by the semantic category of the referent phrase.

Step 4: Statistics and validation. Tab. 7 summarizes UOD-Bench statistics, including the numbers of images, regions, phrases, and forget/retain concepts across tasks and forget ratios.

Note on REC region counts. Unlike OD and PG, REC is formulated as a single-object localization task: each sample contains exactly one target region corresponding to the primary visual referent extracted from the region-level annotations (e.g., "woman with an umbrella" $\rightarrow$ retain only the woman's bounding box). All other regions in the image are discarded. As a result, the number of regions equals the number of images in Tab. 7 for all REC splits.

8.2. Evaluation Protocols and Metrics

This section details the evaluation protocols and metrics used in our experiments, including both in-benchmark evaluations on UOD-Bench and zero-shot evaluations on external datasets.

Within UOD-Bench. For each forget ratio and each task, we report three scores:

- **Forget score** (F): how successfully the model forgets concepts in the forget set.
- **Retain score** (R): how well the model preserves performance on the retain set.
- **U-Score** (U): a harmonic mean between F and R that summarizes the overall unlearning–retention trade-off.

Let M denote the task-specific base metric (mAP@50 for OD, Top-1 with IoU≥ 0.5 for PG, and Top-1 for REC). We define

$$\mathrm{F} = M(\text{forget split}), \quad \mathrm{R} = M(\text{retain split}). \tag{8.1}$$

To balance forgetting efficacy and retention, we compute the U-Score as the harmonic mean of the *forgetting drop* $\Delta_{\text{forget}} = \max(F_{\text{vanilla}} - F_{\text{method}}, 0)$ and the retain performance R_{method}:

$$\mathrm{U} = \frac{2 \cdot \Delta_{\text{forget}} \cdot R_{\text{method}}}{\Delta_{\text{forget}} + R_{\text{method}}}. \tag{8.2}$$

This definition is consistent across tasks and forget ratios, making comparisons between different unlearning methods straightforward.

Zero-shot evaluations. To assess generalization beyond UOD-Bench, we evaluate all models on LVIS-minival and COCO val under the standard open-vocabulary detection protocol [68], *without* explicitly removing forget concepts from these datasets. Together with the Forget/Retain/U-Score defined above on UOD-Bench, these zero-shot evaluations form a multi-granularity protocol that covers both in-benchmark unlearning efficacy and out-of-benchmark generalization, as summarized in Sec. 3 of the main paper.

9. Additional Experiments and Analyses

9.1. Baselines Implementation Notes

This section provides detailed implementation notes for baseline methods in the open-vocabulary object detection (OvOD) unlearning scenario. For a fair comparison, all baselines share the same underlying LLM-Det (Swin-T) / Grounding-DINO (Swin-L) architecture, LoRA configuration ($r = 256$, $\alpha = 512$), optimizer settings (AdamW with weight decay 10^{-4}), and training infrastructure (8$\times$ A800 GPUs with mixed-precision training).

Adaptation to Multi-Task OvOD. In the OvOD + UOD-Bench setting, forget and retain splits are defined at

Table 8. Effect of NPO fine-tuning strategies on UOD-Bench OD task across different forget ratios. F, R, and U denote forget mAP, retain mAP, and U-Score, respectively (lower F and higher R/U are better).

Configuration	1%			5%			10%			15%		
	F ↓	R ↑	U ↑	F ↓	R ↑	U ↑	F ↓	R ↑	U ↑	F ↓	R ↑	U ↑
Vanilla Model	58.0	20.7	-	49.4	20.2	-	37.8	20.7	-	32.6	20.7	-
Head-Only	55.3	21.4	4.8	49.2	20.2	0.4	37.7	19.8	0.2	32.8	20.8	0.0
Head+Decoder	54.5	20.0	6.0	48.2	19.1	2.3	35.1	18.5	4.7	31.6	19.2	1.9
Head+Dec+LM	56.6	20.4	2.6	49.2	18.8	0.4	35.1	18.4	4.7	31.5	19.6	2.1
Extreme	51.4	19.0	9.8	41.0	16.0	11.0	34.5	15.5	5.4	30.5	15.5	3.7
Ultra-Extreme	**50.5**	15.2	**10.0**	**40.0**	14.5	**11.4**	**33.5**	14.1	**6.6**	**29.1**	14.1	**5.6**

the concept level across three tasks (OD, PG, REC). All annotations derived from a forget concept are treated as forget samples, while those from retain concepts are treated as retain samples. Following the adaptation protocol in [11], we extend methods originally designed for text-only (NPO) or single-modal settings (GA, GradDiff) to all three tasks using the unified ODVG data format. Specifically:

- **GA**: We apply gradient ascent on the forget set's detection loss while performing standard gradient descent on the retain set. The ascent coefficient is set to 1.0 following the original paper.
- **NPO**: We adapt the preference optimization objective to the dual-channel dataloader, computing log-ratio preferences between forget and retain predictions on the same image. The temperature β is set to 0.7 with linear warmup.
- **GradDiff**: We compute gradient differences between forget and retain batches and apply the difference as the update direction. The scaling coefficient is set to 0.5 to balance forgetting and retention.
- **MultiDelete**: We implement the saliency-based selective parameter updating strategy by computing Fisher information approximations on forget and retain sets. Parameters with high forget-to-retain saliency ratio (≥ 1.5) are updated, while others are frozen. The mask is recomputed every 100 iterations.

9.2. Fine-Tuning Strategy and Module Selection

We study how trainable module selection affects unlearning performance and establish a unified fine-tuning scope for fair comparison across all methods.

NPO fine-tuning scope ablation. We vary NPO's trainable modules from conservative "Head-Only" (detection head only) through "Head+Decoder" and "Head+Dec+LM" (adding language model blocks) to aggressive "Extreme" (decoder, head, encoder last layer, full BERT; $\beta = 0.5$, 10 epochs) and "Ultra-Extreme" (additionally unfreezing backbone stage-3, neck, encoder last two layers; $\beta = 0.7$, 15 epochs). Tab. 8 reports forget mAP (F), retain mAP (R), and U-Score (U) at four forget ratios.

Conservative settings (Head-Only, Head+Decoder, Head+Dec+LM) barely reduce forget mAP, achieving minimal U-Scores. Extreme substantially improves U-Score by unfreezing deeper modules, while Ultra-Extreme yields the best U-Score among NPO variants. Further unfreezing beyond Ultra-Extreme causes diminishing returns and training instability. We therefore adopt Ultra-Extreme as the unified fine-tuning scope for all methods (GA, NPO, GradDiff, MultiDelete, SafeDetect).

SafeDetect module-wise null-space projection. SafeDetect applies the same Ultra-Extreme module selection but additionally constrains all trainable modules with null-space projection P_{null} to prevent interference with retain concepts. This geometric constraint operates on the decoder-level cross-modal representations (as ablated in main paper Tab. 5), ensuring updates remain orthogonal to the retain subspace while allowing aggressive fine-tuning for effective forgetting.

Connection to deep vs. superficial unlearning. Head-Only's failure (U-Score 4.8 at 1%) and progressive improvement with deeper modules confirm *deep representation-level unlearning* outperforms *superficial output suppression*. From the training scope perspective, Head-Only fails because it only modifies final outputs without changing underlying representations. From the decoupling location perspective (main paper Tab. 5), bbox head decoupling fails because it disrupts strongly aligned pre-trained features. Both observations converge: effective OvOD unlearning requires modifying intermediate, weakly aligned decoder-level representations rather than suppressing classification head outputs.

9.3. Hyperparameter Sensitivity

We analyze the sensitivity of key hyperparameters on UOD-Bench using the LLM-Det Swin-T backbone, reporting results at two representative forget ratios (1%, 15%) to understand their impact on forgetting and retention performance.

Loss weight sensitivity. Tab. 9 shows the effect of varying λ_{flow} and $\lambda_{\text{decouple}}$ on forget/retain performance. At 1% (retaining 99 classes, retain subspace dim ≈95), the balanced configuration $(1.0, 1.0)$ achieves the best U-Score (23.5). At 15% (retaining 85 classes, retain subspace dim ≈82), the larger null-space allows stronger forgetting signals: $(1.5, 1.2)$ achieves U=13.5 vs U=12.9 at $(1.0, 1.0)$. Excessive λ_{flow} (e.g., 2.0) degrades retention (R: 17.2→15.4 at 15%), lowering U to 11.6. We use $\lambda_{\text{flow}} = \lambda_{\text{decouple}} = 1.0$

Table 9. Loss weight sensitivity on UOD-Bench OD task with LLM-Det Swin-T backbone. We test combinations of λ_{flow} and $\lambda_{\text{decouple}}$ at 1% and 15% forget ratios to study their relative importance. The retain subspace dimension is estimated from SVD with threshold $\varepsilon = 10^{-2}$.

		1% Forget Ratio (99 cls, ≈95 dim)			15% Forget Ratio (85 cls, ≈82 dim)		
λ_{flow}	$\lambda_{\text{decouple}}$	F ↓	R ↑	U ↑	F ↓	R ↑	U ↑
0.5	1.0	19.9	15.7	21.2	23.6	16.4	10.6
1.0	1.0	**17.8**	**16.6**	**23.5***	**22.3**	**17.2**	**12.9**†
1.5	1.2	17.3	15.3	22.0	22.0	17.6	13.5*
2.0	1.0	18.1	13.4	18.9	22.1	15.4	11.6

*Best U-Score for this ratio. †Main table configuration (used across all ratios).

Table 10. Null-space threshold ε sensitivity on UOD-Bench OD task with LLM-Det Swin-T backbone. We vary ε in $\{10^{-3}, 10^{-2}, 10^{-1}\}$ while fixing other hyperparameters. Smaller ε leads to a more conservative null-space, while larger ε may under-estimate the retain subspace.

	1% Forget Ratio			15% Forget Ratio		
ε	F ↓	R ↑	U ↑	F ↓	R ↑	U ↑
10^{-3}	22.7	17.6	21.9	25.0	18.1	9.9
10^{-2}	**17.8**	**16.6**	**23.5***	**22.3**	**17.2**	**12.9***
10^{-1}	15.6	15.3	22.9	20.2	12.5	11.2

*Main table configuration.

across all experiments for robust performance.

Null-space threshold sensitivity. The threshold ε in SVD-based null-space construction controls the trade-off between retain coverage and forgetting capacity. Tab. 10 shows that $\varepsilon = 10^{-2}$ achieves the best U-Score at both ratios (23.5 at 1%, 12.9 at 15%). A too-conservative threshold ($\varepsilon = 10^{-3}$) retains nearly all singular values, resulting in an overly large retain subspace that limits the null-space for forgetting (F=22.7 at 1%, F=25.0 at 15%), leading to U-Score degradation to 21.9 and 9.9, respectively. Conversely, a too-aggressive threshold ($\varepsilon = 10^{-1}$) under-estimates the retain subspace and causes severe leakage (R drops to 15.3 at 1%, 12.5 at 15%), despite achieving lower forget scores (F=15.6 at 1%, F=20.2 at 15%). The resulting U-Scores (22.9 at 1%, 11.2 at 15%) are both inferior to the optimal $\varepsilon = 10^{-2}$. Our choice of $\varepsilon = 10^{-2}$ balances both objectives without requiring per-ratio tuning.

Temperature sensitivity. The temperature τ in the mean-flow objective controls the smoothness of the target uniform distribution. Tab. 11 demonstrates that $\tau = 0.07$ provides the best trade-off at both ratios (U=23.5 at 1%, U=12.9 at 15%). Too small τ (0.05) leads to overly sharp distributions that cause optimization instability, resulting in higher F (19.9 at 1%, 24.7 at 15%) and reduced R (16.4 at 1%, 16.5 at 15%), degrading U-Score to 22.1 and 9.3, respectively. Too large τ (0.10) over-smooths the forgetting signal, weakening unlearning effectiveness (F=20.7 at 1%, F=23.4 at 15%) and degrading U-Score to 22.1 and 11.6, respectively. The consistent optimal value across ratios suggests that $\tau = 0.07$ is a robust default choice.

Table 11. Temperature τ sensitivity in the mean-flow forgetting objective on UOD-Bench OD task with LLM-Det Swin-T backbone. We test $\tau \in \{0.05, 0.07, 0.10\}$. Too small τ leads to sharp distributions and unstable optimization, while too large τ weakens the forgetting signal.

	1% Forget Ratio			15% Forget Ratio		
τ	F ↓	R ↑	U ↑	F ↓	R ↑	U ↑
0.05	19.9	16.4	22.1	24.7	16.5	9.3
0.07	**17.8**	**16.6**	**23.5***	**22.3**	**17.2**	**12.9***
0.10	20.7	16.8	22.1	23.4	17.5	11.6

*Main table configuration.

Table 12. LoRA rank r sensitivity on UOD-Bench OD task with LLM-Det Swin-T backbone. We test $r \in \{64, 128, 256, 512\}$ while keeping other hyperparameters fixed. Higher ranks provide more expressive power but incur additional trainable parameters and GPU memory.

	1% Forget Ratio			15% Forget Ratio		
Rank r	F ↓	R ↑	U ↑	F ↓	R ↑	U ↑
64	21.4	16.1	20.8	26.4	16.6	7.5
128	18.0	16.1	22.7	23.2	16.9	11.5
256	**17.8**	**16.6**	**23.5***	**22.3**	**17.2**	**12.9***
512	18.7	16.1	22.3	23.3	16.6	11.1

*Main table configuration.

LoRA rank sensitivity. Tab. 12 shows the effect of LoRA rank r on unlearning performance. At 1%, performance saturates quickly: $r = 128$ achieves U=22.7, and $r = 256$ reaches U=23.5, with further increases to $r = 512$ causing slight degradation (U=22.3) due to overfitting risk. At 15%, the trend is similar: $r = 128$ achieves U=11.5, $r = 256$ reaches the optimal U=12.9, and $r = 512$ shows saturation with slight decline (U=11.1). The consistent saturation pattern across ratios demonstrates that SafeDetect does not require extremely high-rank adaptations to achieve strong unlearning-retention trade-offs, making it practical for resource-constrained scenarios. We use $r = 256$ in the main paper as it provides the best performance at both ratios while maintaining reasonable computational overhead.

9.4. Null-Space Estimation and Retain Coverage

We analyze how the size of the retain set affects null-space estimation and unlearning performance. Tab. 13 reports results when estimating P_{null} from retain subsets of varying sizes (50%, 100%, 200%) while keeping the forget set fixed.

Experimental setup. To understand how retain set size affects null-space estimation, we conduct two complementary studies. First, we vary the coverage of the benchmark retain set (50%, 100%, 200%) and evaluate on UOD-Bench to measure in-benchmark unlearning performance (F/R/U metrics). For coverage below 100%, we subsample the retain set via stratified sampling across the four frequency tiers (Sec. 3), ensuring each tier contributes proportionally

Table 13. Null-space estimation with varying retain coverage on UOD-Bench OD task with LLM-Det Swin-T backbone. We subsample (50%) or augment (200%) the retain set to estimate P_{null} via SVD, while keeping the forget set fixed. Subsampling uses stratified sampling across frequency tiers; augmentation adds semantically similar concepts from the benchmark source vocabularies (COCO+V3Det+GoldG).

			1% Forget Ratio (1 cls)			15% Forget Ratio (15 cls)		
Coverage	#Retain Cls	Est. Dim	F ↓	R ↑	U ↑	F ↓	R ↑	U ↑
50%	49	≈44	16.1	15.7	23.2	20.2	15.9	14.3
100%	99	≈91	**17.8**	**16.6**	**23.5***	**22.3**	**17.2**	**12.9***
200%	198	≈182	18.5	16.5	23.0	22.7	17.0	12.1

*Main table configuration (100% retain coverage).

to preserve semantic diversity. For coverage above 100%, we augment the retain set with semantically similar concepts from COCO, V3Det, and GoldG that are not in the original retain set.

Effect of retain coverage on in-benchmark performance. Results in Tab. 13 show that 50% coverage underestimates the retain subspace, leading to weaker retention at 1% ratio (U-Score 23.2 vs. 23.5 for full coverage). At 15%, 50% coverage yields a higher U-Score (14.3 vs. 12.9) because the smaller retain subspace leaves more null-space for forgetting, but at the cost of lower retain mAP (15.9 vs. 17.2). Full coverage (100%) provides the most consistent retention quality across ratios and is therefore adopted as default. Extending to 200% coverage yields only marginal changes in F/R/U metrics, indicating saturation. Zero-shot evaluation on LVIS-minival (Tab. 14) confirms the same trend, with 100% coverage providing optimal generalization.

Scaling to large external vocabularies. We further test whether expanding the retain vocabulary to large-scale external concepts improves zero-shot generalization on standard benchmarks. We construct larger retain sets (500, 1K, 5K classes) from COCO, V3Det, GoldG, and Objects365 (Tab. 15), then evaluate SafeDetect's zero-shot performance on LVIS-minival without training on UOD-Bench. Results (Tab. 16) reveal a non-monotonic pattern. Expanding from the benchmark's ∼100 classes to 500 classes yields slight improvements due to richer semantic coverage. However, further scaling to 1K and 5K classes causes progressive performance degradation. This behavior aligns with null-space projection theory: moderate expansion enriches the retain subspace without over-constraining the null-space, whereas excessively large vocabularies compress the available null-space and introduce semantic redundancy, ultimately degrading both forgetting efficacy and retention quality.

9.5. Efficiency and Compute Cost

We analyze the computational efficiency of SafeDetect from three perspectives: (1) training efficiency compared to baseline methods, (2) offline SVD cost for null-space construction at different retain vocabulary scales.

Table 14. Impact of retain coverage on zero-shot generalization for SafeDetect on UOD-Bench OD task. We report LVIS-minival zero-shot performance of LLM-Det (Swin-T) under 1% and 15% forget ratios when varying the retain coverage used to estimate P_{null}. Higher is better.

	1% Forget Ratio		15% Forget Ratio	
Coverage	LVIS AP	LVIS AP_r	LVIS AP	LVIS AP_r
50%	35.6	24.6	31.2	20.2
100%	**38.5**	**28.5**	**34.0**	**24.0**
200%	38.4	28.4	33.6	23.7

Main table configuration corresponds to 100% retain coverage.

Table 15. Experimental design for large-scale retain coverage. We expand the benchmark retain set (∼100 classes) to 500, 1K, and 5K safety concepts using COCO, V3Det, GoldG, and Objects365. All configurations exclude forget classes and near-duplicates. For each scale, we estimate P_{null} using all available retain concepts and evaluate SafeDetect under 1% and 15% forget ratios on UOD-Bench OD and zero-shot LVIS-minival.

Retain Scale	#Classes	Construction Strategy
100 (Baseline)	∼100	Benchmark retain set (COCO + task-specific phrases).
500	∼500	COCO base + selected V3Det/GoldG concepts.
1K	∼1,000	500 extended with frequent V3Det categories.
5K	∼5,000	1K extended with most V3Det and Objects365 categories.

Table 16. Impact of large-scale retain coverage on zero-shot generalization. We expand the retain vocabulary from the benchmark-aligned ∼100 classes (99 for 1% ratio, 85 for 15% ratio after removing forget classes) to 500, 1K, and 5K safety concepts following Tab. 15, and report LVIS-minival zero-shot performance.

	1% Forget Ratio (99 cls)		15% Forget Ratio (85 cls)	
Retain Scale	LVIS AP	LVIS AP_r	LVIS AP	LVIS AP_r
Baseline	38.5	28.5	34.0	24.0
500	38.6	28.6	34.2	24.3
1K	38.1	28.2	33.6	23.7
5K	37.8	28.0	33.6	23.6

Training efficiency comparison. Tab. 17 compares SafeDetect against NPO on UOD-Bench with a 1% forget ratio. All measurements are conducted on the same hardware (8×A100-SXM4-80GB GPUs) using identical implementations. SafeDetect achieves 40% fewer GPU-hours (13.2 vs 22.0) compared to NPO, while maintaining comparable algorithm FLOPs (54.60 GFLOPs vs ∼54 GFLOPs). The efficiency gain comes from faster convergence enabled by the geometric null-space projection, which provides more stable gradient updates and better retention protection. Despite the additional one-time SVD cost for constructing the null-space projector, the overall training cost remains significantly lower than NPO.

Offline SVD cost at different scales. Tab. 18 reports the measured wall-clock time and FLOPs for SVD decomposition at different retain vocabulary scales (100, 500, 1K, 5K classes). Each measurement is averaged over 10 runs after GPU warm-up. The SVD cost grows sub-linearly with vocabulary size due to rank saturation: the output rank is capped at the text embedding dimension $d = 768$, so expanding from 1K to 5K classes only increases the time

(a) Vanilla Model (b) NPO (c) Ours

Referential expression comprehension: ***the woman touching the fountain water***

Figure 8. Qualitative comparison on REC for the query *"the woman touching the fountain water"*. **(a)** Vanilla model correctly understands the expression and localizes the target woman. **(b)** NPO fails to accurately forget the target (red circle), leaving residual detections. **(c)** Our method achieves precise unlearning, completely removing the target while avoiding spurious changes to the scene.

Table 17. Comparison of SafeDetect (Ours) against NPO on training efficiency (1% forget ratio, UOD-Bench OD task). All measurements are conducted on 8×A100-SXM4-80GB GPUs with identical implementations.

Metric	NPO	SafeDetect (Ours)	Improvement
Algorithm FLOPs	∼54 GFLOPs	54.60 GFLOPs	Comparable
GPU-hours ↓	22.0	**13.2**	**40% fewer**
Trainable Params	831M	831M	Same

Table 18. Offline SVD computational cost for constructing the retain subspace at different vocabulary scales. We report measured wall-clock time and FLOPs for a single SVD decomposition on an NVIDIA A100-SXM4-80GB GPU. Text embedding dimension is $d = 768$. Each measurement is averaged over 10 runs after GPU warm-up. The cost grows sub-linearly due to rank saturation (output rank capped at $d = 768$).

Retain Scale	#Classes N	Embedding Matrix Size	SVD Time (per module)	FLOPs (per module)
100 (Baseline)	100	100×768	0.01s	0.015 GFLOPs
500	500	500×768	0.03s	0.384 GFLOPs
1K	1,000	$1{,}000 \times 768$	0.05s	1.180 GFLOPs
5K	5,000	$5{,}000 \times 768$	0.06s	5.898 GFLOPs

from 0.05s to 0.06s. Even for the largest 5K-class vocabulary, the one-time SVD cost is negligible (0.06s, 5.898 GFLOPs) compared to the total training time (1.65 hours). This demonstrates that SafeDetect's geometric approach is highly scalable and practical for real-world deployment scenarios requiring large retain vocabularies.

9.6. Generalization of Forgetting to External Benchmarks

In this section, we examine whether unlearning trained on UOD-Bench generalizes to external zero-shot benchmarks, specifically testing whether forgotten concepts remain suppressed and whether unrelated concepts are preserved on LVIS-minival. We compute the semantic overlap between the 15 forget concepts at 15% ratio and LVIS categories, obtaining 12 overlapping classes (1.0% of 1,203 total). We partition LVIS into overlapping (forget-related) and non-overlapping (unrelated) subsets, and report mAP@50 for

Table 19. Adversarial robustness under four attack types at 1% and 15% forget ratios. ASR closer to 0.5 is better for membership inference; lower recovery rate and higher relearning steps indicate stronger unlearning.

Attack	Vanilla		NPO		MUNBa		SafeDetect (Ours)	
	1%	15%	1%	15%	1%	15%	1%	15%
Membership Inference (ASR↓, 0.5 = random guess)								
LiRA [4]	0.93	0.86	0.74	0.67	0.67	0.63	**0.62**	**0.58**
VL-MIA	0.88	0.84	0.69	0.65	0.65	0.57	**0.58**	**0.55**
Concept Recovery (Rate↓) / Relearning (Steps↑)								
Unlearn-Inv	–	–	0.64	0.56	0.55	0.48	**0.43**	**0.39**
Relearning	0	0	45	36	61	47	**92**	**68**

Vanilla (no unlearning), w/o Null-space, and SafeDetect in Tab. 22. SafeDetect achieves strong suppression on overlapping classes (48.1→25.8, −46.4% vs. Vanilla), while maintaining higher non-overlapping AP (39.1 vs. 34.3 for w/o Null-space), demonstrating that null-space projection effectively removes targeted concepts while substantially reducing damage to unrelated categories.

9.7. Adversarial Robustness Evaluation

A practical unlearning system must resist attempts to recover erased knowledge. We evaluate SafeDetect against four attack types covering membership inference, concept recovery, and relearning, comparing against Vanilla (no unlearning), NPO [65], and MUNBa [53] at 1% and 15% forget ratios (Tab. 19).

Membership inference. LiRA [4] and VL-MIA exploit training-distribution signals to infer whether a concept was seen during training. A lower attack success rate (ASR) is better, with 0.5 corresponding to random guess.

Concept recovery and relearning. Unlearn-Inv [21] inverts unlearning to reconstruct forgotten detections, where a lower recovery rate is better. Relearning attacks measure fine-tuning steps needed to restore forgotten performance, with more steps indicating stronger forgetting.

SafeDetect achieves MIA ASR closest to random (0.62/0.58 at 1%/15%), lowest concept recovery

(0.43/0.39), and highest relearning steps (92/68 — 2× NPO). The null-space constraint geometrically removes concepts from weight space instead of suppressing output, hindering recovery.

9.8. Comparison with Image-Domain and VLM Unlearning Methods

To comprehensively evaluate the effectiveness of SafeDetect, we extend our study to compare with representative machine unlearning (MU) methods from both the image domain and vision-language models (VLMs). While these methods are primarily designed for closed-set classification or global cross-modal alignment, we carefully adapt their optimization objectives to the OvOD setting to assess their applicability.

Comparison with Image-Domain MU. We evaluate four widely used image-domain unlearning methods: Influence [25], SalUn [13], GDR-GMA [31], and MUNBa [53]. As shown in Table 20, these methods struggle to effectively balance forgetting and retention. Because they do not account for the decomposable structure of VLM embeddings, their parameter updates inevitably leak into shared semantic directions, leading to either insufficient forgetting or severe degradation of retained concepts.

Table 20. Comparison with image-domain MU methods on UOD-Bench OD task (LLM-Det Swin-T). F/R/U denote forget mAP, retain mAP, and U-Score. Lower F and higher R/U are better.

Method	**1% Forget**			**15% Forget**		
	F↓	R↑	U↑	F↓	R↑	U↑
Vanilla Model	58.0	20.7	–	32.6	20.7	–
NPO [65]	50.5	15.2	10.0	29.1	14.1	5.6
Influence [25]	55.5	**18.0**	4.4	31.5	**17.5**	2.1
SalUn [13]	52.0	17.0	8.9	30.5	16.5	3.7
GDR-GMA [31]	48.0	16.5	12.5	29.5	16.0	5.2
MUNBa [53]	45.0	16.0	14.3	28.5	15.5	6.5
SafeDetect (Ours)	**17.8**	16.6	**23.5**	**22.3**	17.2	**12.9**

Comparison with VLM Unlearning Methods. We further compare with recent VLM concept erasure methods, including CLIPErase [60] and CAGUL [2]. We exclude EraseDiff [54] as its generative score-function manipulation is structurally incompatible with discriminative detection. Table 21 demonstrates that while VLM methods directly manipulate cross-modal representations and generally perform better than image-domain MU, they still significantly under-forget. They are not designed to handle region-level localization or the strongly entangled decoder representations inherent in dense OvOD models.

SafeDetect outperforms all adapted image-domain and VLM unlearning baselines by a clear margin, achieving a +7.4 U-Score improvement over the best VLM method (CLIPErase) at the 1% forgetting ratio. These results show that our geometrically constrained null-space projection, which provides a strict theoretical guarantee against geometric entanglement, is far more effective than soft loss balancing when removing targeted concepts in OvOD.

Table 21. Comparison with VLM unlearning methods adapted to OvOD on UOD-Bench OD task (LLM-Det Swin-T).

Method	**1% Forget**			**15% Forget**		
	F↓	R↑	U↑	F↓	R↑	U↑
Vanilla Model	58.0	20.7	–	32.6	20.7	–
CLIPErase [60]	42.0	16.2	16.1	27.5	15.8	7.7
CAGUL [2]	44.5	16.8	15.0	28.0	16.2	7.2
SafeDetect (Ours)	**17.8**	16.6	**23.5**	**22.3**	17.2	**12.9**

Table 22. Zero-shot forget behavior on LVIS-minival (15% forget ratio). We partition LVIS into overlapping classes (12 categories semantically matching UOD-Bench forget concepts via exact matches and synonyms) and non-overlapping classes (1,191 unrelated concepts), then report LVIS AP on each subset. SafeDetect achieves strong suppression on overlapping classes (-46.4% vs Vanilla) while better retaining non-overlapping performance ($+14.0\%$ vs w/o Null-space).

Method	Overlap (12 cls)	Non-Overlap (1191 cls)
Vanilla (No Unlearning)	48.1	44.6
w/o Null-space	33.8	34.3
SafeDetect (Ours)	**25.8**	**39.1**

10. Additional Visualizations

REC qualitative behavior. Fig. 8 provides a qualitative comparison on the REC task for the query *"the woman touching the fountain water"*. In the Vanilla model (Fig. 8a), the detector correctly understands the referring expression and localizes the target woman. NPO (Fig. 8b) fails to reliably erase this concept, leaving residual detections of the woman even after unlearning, illustrating incomplete forgetting on a privacy-sensitive target. Our SafeDetect method (Fig. 8c) completely removes the woman while leaving the surrounding scene and non-forget entities unchanged, demonstrating precise concept removal without collateral distortion in REC.

11. Failure Cases and Limitations

We identify two primary failure modes of SafeDetect. **(1) Near-synonym leakage.** When forget and retain concepts are near-synonyms (*e.g.*, "person" vs. "human"), their embeddings are nearly collinear, leaving minimal null-space for forgetting updates. This is a fundamental boundary of concept-level unlearning: the geometric constraint that protects retain directions inherently limits erasure of semantically overlapping concepts. **(2) High forget-ratio degradation.** At aggressive forget ratios ($\geq$15%), the null-space dimensionality shrinks as more concept directions

are removed, reducing the effective capacity for forgetting updates. Both limitations stem from the geometric constraints that ensure retention safety, suggesting a fundamental forgetting–retention Pareto frontier in null-space-based unlearning.

12. Theoretical Foundations and Proofs

12.1. Preliminaries and Notation

We align notation with the main paper and only restate essentials. Decomposable embeddings (see Main Paper Def. 1): VLM text embeddings admit an approximately decomposable form $\bar{\ell}_z \approx \bar{\ell}_0 + \sum_i \bar{\ell}_{z_i}$, which induces shared semantic factors across concepts in the aligned space $\mathcal{S}_{\text{OvOD}}$.

Shapes and matrices. - Let d denote the text embedding dimension. For a retain concept set of size k, the retain embedding matrix is $\mathbf{F}_r = [\mathbf{f}_{c_1}, \ldots, \mathbf{f}_{c_k}] \in \mathbb{R}^{d\times k}$, with columns $\mathbf{f}_c \in \mathbb{R}^d$. - We write W for the model's linear operator (or the effective linearization/Jacobian at the interface layer) that maps text embeddings to alignment scores/logits where region–text alignment is evaluated, so that expressions like $W \cdot \mathbf{f}_c$ denote these alignment scores.

Left null-space and constraint. For a matrix $\mathbf{A} \in \mathbb{R}^{d\times k}$, a matrix (or vector) B is in the *left null-space* of $\mathbf{A}$ if $B\mathbf{A} = \mathbf{0}$. In our context, an update ΔW satisfies the null-space constraint if

$$\Delta W \cdot \mathbf{F}_r = \mathbf{0} \quad \Longleftrightarrow \quad \langle \Delta W, \mathbf{f}_c \rangle = 0,\ \forall\, \mathbf{f}_c \in \mathbf{F}_r,$$

which enforces orthogonality to all retain directions.

Projectors. Let $\mathbf{F}_r = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ be an SVD and let $\mathbf{U}_r$ collect singular vectors above a stability threshold. The retain projector is $P_{\text{keep}} = \mathbf{U}_r\mathbf{U}_r^\top$ and the null-space projector is $P_{\text{null}} = I - P_{\text{keep}}$. Any direction decomposes as $\Delta W = P_{\text{keep}}\Delta W + P_{\text{null}}\Delta W$, where the second term is orthogonal to $\text{span}(\mathbf{F}_r)$.

These notions will be used to formalize retain invariance and first-order interference elimination below.

12.2. Null-Space Safety: Retain Invariance and First-Order Interference Elimination

We provide the complete proof of Proposition 1 (Retain Invariance via Null Space) from the main paper, including detailed geometric entanglement analysis.

Proposition 1 (Retain Invariance via Null Space - Full Statement)**.** *Let ΔW satisfy the null space constraint:*

$$\langle \Delta W, \mathbf{f}_c \rangle = 0, \quad \forall \mathbf{f}_c \in \mathbf{F}_r. \tag{12.1}$$

For any retain concept $c \in \mathcal{C}_{retain}$ with embedding $\mathbf{f}_c$, the region-text alignment remains unchanged:

$$(W + \Delta W) \cdot \mathbf{f}_c = W \cdot \mathbf{f}_c. \qquad \text{(Main Paper Eq. 4.1)}$$

Consequently, the geometric entanglement interference is eliminated: for all $c \in \mathcal{C}_{retain}$, $\Delta^{(1)}align(c) = 0$.

Proof. **Step 1: Geometric Entanglement Analysis.** Recall from Definition 1 (Decomposable Embeddings) in the main paper that VLM embeddings exhibit linear compositionality:

$$\bar{\ell}_z = \bar{\ell}_0 + \sum_{i=1}^{k} \bar{\ell}_{z_i}, \tag{12.2}$$

where concepts share decomposed semantic factors in $\mathcal{S}_{\text{align}}$. This linear structure implies that concepts with overlapping factors (e.g., "woman", "person", "child") are *geometrically entangled* in the embedding space.

In unconstrained unlearning, forgetting updates ΔW_f aimed at erasing forget concepts are computed without geometric constraints. Due to geometric entanglement, these updates possess non-zero projections onto the retain subspace $\text{span}(\mathbf{F}_r)$:

$$\text{Proj}_{\text{span}(\mathbf{F}_r)}(\Delta W_f) = \sum_i \frac{\langle \Delta W_f, \mathbf{f}_{r_i} \rangle}{\|\mathbf{f}_{r_i}\|^2} \mathbf{f}_{r_i} \neq \mathbf{0}. \tag{12.3}$$

This non-zero projection perturbs the shared semantic factors between forget and retain concepts, causing geometric entanglement interference (Main Paper Eq. 5):

$$\Delta^{(1)}\text{align}(c) = \langle \Delta W_f, \mathbf{f}_c \rangle \neq 0, \quad \text{for some } c \in \mathcal{C}_{\text{retain}}. \tag{12.4}$$

Step 2: Null Space Constraint Enforcement. Our framework eliminates this interference by enforcing the null space constraint:

$$\langle \Delta W, \mathbf{f}_c \rangle = 0, \quad \forall \mathbf{f}_c \in \mathbf{F}_r. \tag{12.5}$$

This constraint mathematically guarantees that the update ΔW is orthogonal to all retain embeddings, effectively residing in the null space of the retain subspace.

Step 3: Retention Guarantee. Expanding the alignment term for the updated parameters $(W + \Delta W)$ with any retain embedding $\mathbf{f}_c$:

$$(W + \Delta W) \cdot \mathbf{f}_c = W \cdot \mathbf{f}_c + \Delta W \cdot \mathbf{f}_c = W \cdot \mathbf{f}_c + 0 = W \cdot \mathbf{f}_c, \tag{12.6}$$

where the second term $\Delta W \cdot \mathbf{f}_c = 0$ vanishes by the null space constraint. This proves that the region-text alignment for any retain concept remains exactly unchanged after the update.

Step 4: Interference Elimination. From the null space constraint, the first-order interference term becomes:

$$\Delta^{(1)}\text{align}(c) = \langle \Delta W, \mathbf{f}_c \rangle = 0, \quad \forall c \in \mathcal{C}_{\text{retain}}. \tag{12.7}$$

Furthermore, the projection of ΔW onto the retain subspace from Eq. 12.3 becomes:

$$\text{Proj}_{\text{span}(\mathbf{F}_r)}(\Delta W) = \sum_i \frac{\langle \Delta W, \mathbf{f}_{r_i} \rangle}{\|\mathbf{f}_{r_i}\|^2} \mathbf{f}_{r_i} = \sum_i \frac{0}{\|\mathbf{f}_{r_i}\|^2} \mathbf{f}_{r_i} = \mathbf{0}, \tag{12.8}$$

where each inner product $\langle \Delta W, \mathbf{f}_{r_i} \rangle = 0$ by the null space constraint. This proves that the geometric entanglement interference is completely eliminated.

Step 5: Zero-Shot Generalization Preservation. The elimination of geometric entanglement interference has a critical consequence: it preserves the geometric structure of $\mathcal{S}_{\text{align}}$ not only for retain concepts but also for the entire embedding space. This ensures that zero-shot generalization capability is maintained for unseen concepts beyond both $\mathcal{C}_{\text{forget}}$ and $\mathcal{C}_{\text{retain}}$.

Consider an unseen concept c_u whose embedding $\mathbf{f}_{c_u}$ shares decomposed factors with retain concepts. Since ΔW is orthogonal to all retain embeddings in $\mathbf{F}_r$, and $\mathbf{f}_{c_u}$ can be decomposed as:

$$\mathbf{f}_{c_u} = \mathbf{f}_0 + \sum_{i \in I_r} \alpha_i \mathbf{f}_{r_i} + \sum_{j \in I_u} \beta_j \mathbf{f}_{u_j}, \tag{12.9}$$

where I_r indexes shared factors with retain concepts and I_u indexes unique factors, the alignment change becomes:

$$\begin{aligned} \Delta^{(1)}\text{align}(c_u) &= \langle \Delta W, \mathbf{f}_{c_u} \rangle \\ &= \sum_{i \in I_r} \alpha_i \underbrace{\langle \Delta W, \mathbf{f}_{r_i} \rangle}_{=0} + \sum_{j \in I_u} \beta_j \langle \Delta W, \mathbf{f}_{u_j} \rangle. \end{aligned} \tag{12.10}$$

The shared factors with retain concepts contribute zero change, preserving the geometric structure for generalization. □

Corollary (Shared-Factor Preservation in Decomposable Embeddings). Assume text embeddings admit the decomposable structure in Def. 2.1 and let $\text{span}(\mathbf{F}_r)$ denote the retain subspace. If ΔW satisfies $\langle \Delta W, \mathbf{f}_c \rangle = 0$ for all $\mathbf{f}_c \in \mathbf{F}_r$, then for any unseen concept with

$$\mathbf{f}_{c_u} = \mathbf{f}_0 + \sum_i \alpha_i \, \mathbf{f}_{r_i} + \sum_j \beta_j \, \mathbf{f}_{u_j}, \quad \mathbf{f}_{r_i} \in \text{span}(\mathbf{F}_r),$$

we have $\sum_i \alpha_i \langle \Delta W, \mathbf{f}_{r_i} \rangle = 0$. Hence the components shared with retain factors are preserved in the first order, implying invariance of shared directions.

12.3. Analysis of One-Step Detection Unlearning

In this subsection, we clarify how the one-step unlearning loss in the main paper,

$$\mathcal{L}_{\text{flow}}^{(\mathcal{D}_f)} = \mathbb{E}_{x \in \mathcal{D}_f} \text{KL}\big(\text{softmax}(\mathbf{z}_\theta(x)/\tau), \mathcal{U}\big), \tag{12.11}$$

relates to mean-flow style objectives [17, 33] and why it is effective for forgetting. For logits $z \in \mathbb{R}^k$ over $k = |\mathcal{C}_{\text{forget}}|$ classes, let $p = \text{softmax}(z/\tau)$ and $U = \frac{1}{k}\mathbf{1}$. Then

$$\mathcal{L}_{\text{flow}}(z) = \text{KL}(p, U) = \sum_{i=1}^{k} p_i \log \frac{p_i}{1/k}, \tag{12.12}$$

whose gradient w.r.t. logits has the simple form

$$\frac{\partial \mathcal{L}_{\text{flow}}}{\partial z} = \frac{1}{\tau}(p - U). \tag{12.13}$$

Thus a gradient step with step size η updates logits as

$$z^+ = z - \eta \frac{(p - U)}{\tau}, \tag{12.14}$$

directly shrinking the deviation $p - U$ from the uniform target on the probability simplex.

This mirrors the spirit of MeanFlow: instead of explicitly integrating an ODE over many steps, a single learned "average-velocity" step moves samples toward the terminal distribution. Here, the KL-to-uniform objective plays the role of an average flow on the logit space: each update contracts confidence gaps among forget classes, monotonically reducing over-confident predictions and driving them toward the maximum-entropy state $p \approx \mathcal{U}$. This explains the fast, non-oscillatory convergence of forgetting curves in Main Paper Fig. 5 compared to multi-step objectives.

12.4. Analysis of Decoder-Level Cross-Modal Decoupling

We analyze the cross-modal decoupling loss (Main Paper Eq. 4.8) to explain Obs. 5 and Tab. 5: *why decoder-level decoupling achieves superior forgetting-retention trade-offs compared to bbox-head decoupling*.

Setup. Decoder queries $\{v_i\}$ are intermediate representations for object localization. Text embeddings $\{f_j\}$ span the aligned semantic space. Their similarity matrix is $\mathbf{S}_{ij} = \text{sim}(v_i, f_j)/\tau$. Before unlearning, $\mathbf{S} \approx 0$ (Main Paper Fig. 7), indicating *weak alignment*.

Decoupling loss. The objective

$$\mathcal{L}_{\text{decouple}} = \tfrac{1}{2}\big[\ell_{\text{CE}}(-\mathbf{S}, \mathbf{I}) + \ell_{\text{CE}}(-\mathbf{S}^\top, \mathbf{I})\big] \tag{12.15}$$

pushes each forget query v_i away from its matched text embedding f_i while keeping it orthogonal to others.

Gradient analysis. Let $\pi_i = \text{softmax}(-\mathbf{S}_{i\cdot})$. For the row-wise cross-entropy term $\ell_{\text{CE}}(-\mathbf{S}_{i\cdot}, \mathbf{e}_i) = -\log \pi_{ii}$, we have

$$\frac{\partial \mathcal{L}_{\text{decouple}}}{\partial S_{ij}} = \pi_{ij} - \mathbf{I}_{ij}. \tag{12.16}$$

At initialization $\mathbf{S} \approx 0$, the softmax is nearly uniform ($\pi_{ij} \approx 1/k$), so

$$\frac{\partial \mathcal{L}_{\text{decouple}}}{\partial S_{ij}} \approx \begin{cases} -(1 - \frac{1}{k}), & i = j, \\ \frac{1}{k}, & i \neq j. \end{cases} \tag{12.17}$$

Since $S_{ij} = \text{sim}(v_i, f_j)/\tau$ and $\partial S_{ij}/\partial v_i \propto f_j$, the feature-space gradient on v_i is *a bounded step along $-f_i$ with small leakage to other f_j*.

Weak-alignment regime enables short path with small conflict. Because v_i starts near zero similarity, moving from $S_{ii} \approx 0$ to moderately negative values (-0.6 to -0.8 in Main Paper Fig. 7) carves out a local repulsive cone around forget directions. This short path operates in the decoder's intermediate space, where the projection onto retain directions span($\mathbf{F}_r$) remains small. As a result, *decoder-level decoupling induces minimal gradient conflicts with retain concepts*.

Bbox-head regime suffers from long path with large conflict. In contrast, bbox classification logits are already strongly aligned with text embeddings (high confidence). Applying repulsion here requires inverting large positive logits to negative values. The resulting gradients backpropagate through large classifier weights into all feature channels shared by forget and retain concepts. This produces large tangential components (Eq. 4.3), amplifying geometric entanglement interference and causing the substantial retain mAP and U-Score drops observed in Main Paper Tab. 5.

Conclusion. Decoder-level decoupling implements *deep, representation-level unlearning* by operating in a weakly aligned regime with bounded gradients and minimal interference to the retain subspace. Bbox-head decoupling, by contrast, performs *superficial output-space inversion* that disrupts shared feature channels and degrades retention. This gradient-conflict perspective explains why targeting intermediate representations is safer and more effective than suppressing final outputs.

12.5. Assumptions and Limitations

Our theoretical guarantees rely on the following modeling assumptions:

- **Decomposable embeddings.** We assume OvOD text representations follow approximate linear compositionality with bounded residuals (Main Paper Definition 1). This underpins the intuition that protecting retain factors also preserves zero-shot neighbors. This assumption is supported by recent work on compositional VLM representations and model editing [14, 49].
- **Finite retain set for null-space estimation.** The null-space projector P_{null} is estimated from a finite retain set $\mathbf{F}_r$. When coverage is incomplete, the retain-invariance guarantee (Proposition 1) becomes approximate. However, our retain-coverage ablations (Sup. Tab. 13) show that 100% benchmark coverage achieves strong empirical performance, and the guarantee degrades gracefully with reduced coverage.
- **Contraction regime for mean-flow objective.** The one-step KL-to-uniform forgetting objective assumes detector logits on forget classes lie in a regime where the gradient induces contraction toward uniform predictions. Extremely skewed class imbalance or heavy label noise may slow convergence but do not invalidate the objective.

Our experiments across four forgetting ratios (1%-15%) and three tasks (OD/PG/REC) demonstrate robust performance under diverse data distributions.