WHICH NEGATIVES MATTER? ASK YOUR TEXT ENCODER: ADAPTIVE SIMILARITY MARGINS FOR DENSE-CAPTION RETRIEVAL

Anonymous authors
Paper under double-blind review

ABSTRACT

Dense-caption retrieval requires distinguishing images paired with highly similar, detail-rich descriptions. We empirically examine contrastive fine-tuning from a strong pretrained dual encoder and observe that the standard InfoNCE loss can become small on sampled batches while held-out retrieval errors remain. Through an analysis of gradient dynamics, we find that InfoNCE rapidly loses optimization signal: within the first epoch, its loss falls below 10^{-3} on 80% of sampled batches, and over the run its gradient norm below 5×10^{-7} in 47% of measurements, even as held-out retrieval continues to improve. We relate this premature saturation to the abundance of near-duplicate captions in dense-caption datasets, where the most confusable negatives remain unresolved. We therefore introduce HN-CLIP, which uses the text encoder's own text–text geometry to construct per-negative adaptive similarity margins. A detached caption-similarity matrix is added to the negative logits, assigning larger margins to more similar captions without mining, synthesizing, or resampling negatives. The objective requires only one similarity matrix and a masked logit addition during training, with no auxiliary data, additional parameters, offline preprocessing, or inference-time overhead. Across four dense-caption retrieval benchmarks, HN-CLIP improves over the strongest competitors by $+2.5$–$+4.0$ Recall@1 while training $2.4\times$ faster than GOAL and $5.4\times$ faster than StructXLIP. It also improves all six tested fine-tuning frameworks in-domain and surpasses full-data GOAL and StructXLIP with only 20% of the DOCCI and DCI training data, demonstrating strong data efficiency.

1 INTRODUCTION

Dense-caption retrieval is a fine-grained ranking problem in which visually similar images often share overlapping objects, scenes, and actions, while subtle attribute or relation differences determine the correct match, as seen in DOCCI (Onoe et al., 2024), DCI (Urbanek et al., 2024), and Urban-1K (Zhang et al., 2024). Recent methods build on contrastively pre-trained vision–language models (Radford et al., 2021; Zhai et al., 2023) and Long-CLIP (Zhang et al., 2024), enriching training with fine-grained cross-modal alignment, region-level matching, adaptive masking, or structural cues (Asokan et al., 2025; Choi et al., 2025; Xie et al., 2025; Ruan et al., 2026). These advances raise a complementary question: when a confusable caption appears among the negatives, how much separation is needed before its contribution to the global contrastive loss becomes negligible?

We therefore investigate whether this objective remains effective during dense-caption fine-tuning. Our analysis reveals a pronounced saturation phenomenon: in a diagnostic run from Long-CLIP, the InfoNCE loss falls below 10^{-3} on 80% of measured batches within the first epoch, and its gradient norm falls below 5×10^{-7} in 47% of measurements (Section 2.4). We trace this behavior to the caption geometry of dense-caption datasets. Long descriptions are highly compositional and often form near-duplicate pairs: under the pre-trained Long-CLIP text encoder, the mean hardest-negative similarity is 0.92–0.94 (Fig. 1). A strong initialization therefore separates the easy majority of negatives very quickly, so sampled batches appear solved while the highly similar candidates that decide Recall@1 remain unresolved in the full gallery. This observation raises a natural question: *can the contrastive objective itself adapt its training pressure to the hard negatives already present in dense-caption batches?*

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

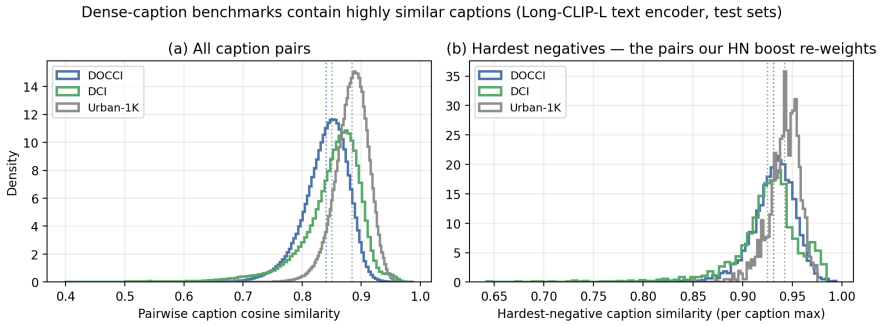

Figure 1: **Dense-caption datasets are dominated by hard negatives.** Distributions of (a) all pairwise caption-caption cosine similarities and (b) each caption's hardest-negative similarity, measured with the pre-trained Long-CLIP-L text encoder on the test sets. Dotted lines mark the means. Higher caption similarity corresponds to a larger boost when a pair appears in a training batch.

To address this question, we introduce HN-CLIP, a simple objective-level approach that exploits the text encoder's own representation geometry to identify hardness. Given a batch of image–caption pairs, HN-CLIP computes the caption–caption similarity matrix, detaches it from the computation graph, masks its diagonal, and adds it to the negative logits with a single coefficient γ. Each negative thus receives an adaptive margin proportional to its caption similarity: less similar negatives receive smaller shifts, whereas near-duplicate captions require larger separation before their loss becomes negligible. Unlike conventional hard-negative strategies, HN-CLIP does not mine, synthesize, or resample negatives; the existing batch is left unchanged. Combined with a standard token-level late-interaction term (Yao et al., 2021), the method introduces no auxiliary inputs, preprocessing pipeline, architectural module, additional parameters, or inference-time computation.

Our contributions are summarized as follows:

- **We first show how a seemingly solved batch can conceal an unsolved dense-caption retrieval problem.** In a gradient-dynamics study of Long-CLIP-L, standard InfoNCE, evaluated along a HN-CLIP training trajectory, becomes nearly inactive on most sampled batches while held-out retrieval errors persist. Caption-similarity analysis further shows that full retrieval galleries contain highly similar descriptions. Together, these observations motivate a loss that continues to challenge confusable negatives when they appear in a batch.
- **We introduce HN-CLIP, a simple adaptive-margin objective for dense-caption retrieval.** HN-CLIP converts the text encoder's own text–text geometry into detached, per-negative similarity margins, assigning stronger training pressure to more similar captions without negative mining, synthesis, resampling, auxiliary data, architectural changes, or inference overhead.
- **We provide extensive empirical evidence for the effectiveness and generality of the proposed objective.** Across four dense-caption benchmarks, HN-CLIP leads all eight retrieval directions in Table 1, with gains of $+2.5$–$+4.0$ over its best competitors. It trains faster than GOAL and StructXLIP, surpasses those full-data competitors using one fifth of the DOCCI and DCI training data, and improves all six tested fine-tuning frameworks on the in-domain benchmarks. Gradient analysis further connects these improvements to sustained optimization signal on hard negatives.

2 METHOD

In this section, we introduce HN-CLIP, which fine-tunes a pre-trained dual encoder without auxiliary data views, preprocessing pipelines, or architectural modules (Figs. 2 and 3). Training combines a text-similarity–boosted global objective with a standard token-level late-interaction term: (1) The *text-similarity–boosted objective* (Section 2.2), which converts the batch's own caption-similarity matrix into per-negative adaptive margins, keeping the loss and its gradient active on confusable pairs; and (2) A *token-level alignment branch* (Section 2.3) that complements the global boost, with additional gains on Long-DCI.

We first quantify why the standard objective fails on dense captions (Section 2.1), then present both components, and close with an empirical gradient-dynamics analysis (Section 2.4).

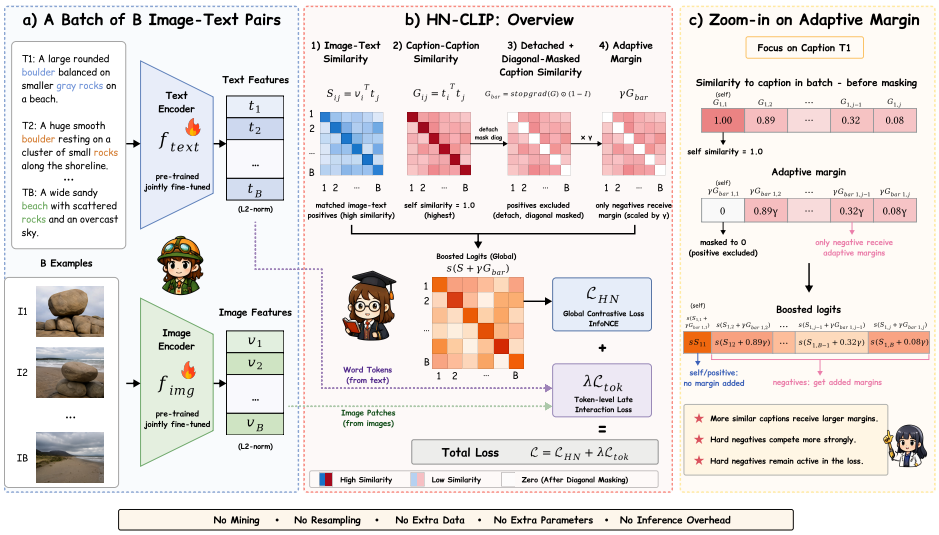

Figure 2: **Overview of HN-CLIP. (a)** A batch of image–text pairs is encoded by the jointly fine-tuned dual encoder. **(b)** Caption–caption similarity provides detached, per-negative adaptive margins for the global contrastive loss, combined with the token-level term. **(c)** More similar negatives receive larger margins, while the positive is unchanged. HN-CLIP requires no mining, resampling, extra data, additional parameters, or inference-time overhead.

2.1 Preliminaries and Motivation

Let f_{img} and f_{txt} map images and captions into a shared D-dimensional space, and let $\{(I_i, T_i)\}_{i=1}^{B}$ denote a batch of B image-caption pairs. Let $\mathbf{v}_i = f_{\text{img}}(I_i)/\|f_{\text{img}}(I_i)\|$ and $\mathbf{t}_i = f_{\text{txt}}(T_i)/\|f_{\text{txt}}(T_i)\|$ denote the normalized embeddings, $S_{ij} = \mathbf{v}_i^\top \mathbf{t}_j$ the image-text similarity matrix, and s the learned inverse temperature. Standard fine-tuning minimizes the symmetric InfoNCE loss (Oord et al., 2018; Radford et al., 2021)

$$\mathcal{L}_{\text{CLIP}} = \tfrac{1}{2}\big[\text{CE}(sS, y) + \text{CE}(sS^\top, y)\big], \quad (1)$$

where $\text{CE}(Z, y)$ denotes row-wise softmax cross-entropy with target indices $y_i = i$. Writing ℓ_i^0 for the i-th row term of Eq. (1), its similarity-score gradient is the classic softmax residual (Wang & Liu, 2021), $\partial \ell_i^0 / \partial S_{ij} = s\, p_{ij}$ for $j \neq i$, with $p_{ij} = [\text{softmax}(sS_{i,:})]_j$: each negative is repelled with force proportional to its posterior. When a strong pretrained initialization such as Long-CLIP separates the positive from the sampled negatives, these posteriors and the loss become small, and the resulting gradient can become vanishingly small, while retrieval errors persist in the full gallery.

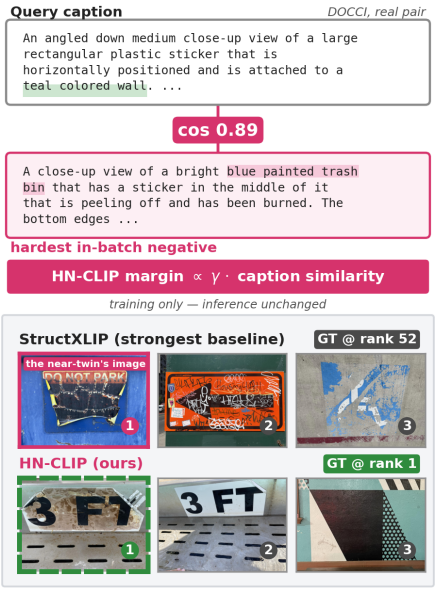

Figure 3: **Illustration of HN-CLIP.** A real DOCCI query and its similar negative (cos 0.89). StructXLIP ranks the ground truth 52nd and retrieves the near-twin's image first (pink); HN-CLIP ranks the ground truth first.

Dense-caption datasets expose a potential mismatch between batch-level optimization and full-gallery evaluation. Using the released Long-CLIP-L text encoder, mean pairwise caption similarities on DOCCI, DCI, and Urban-1K are 0.84, 0.85, and 0.88, while nearest-nonmatching similarities average 0.93, 0.92, and 0.94 (Fig. 1). These statistics show that dense-caption galleries contain many highly similar descriptions, often differing only in a few attributes or relations. Training, however, compares each positive only against negatives in the current batch, so a batch can attain a very small InfoNCE loss after separating easy negatives while more confusable candidates remain in the full gallery. We verify this mismatch through loss, gradient, and retrieval-error dynamics in Section 2.4, motivating caption similarity as an additional in-batch difficulty signal.

2.2 TEXT-SIMILARITY–BOOSTED HARD NEGATIVES

The diagnosis suggests the remedy: the objective must know *which* negatives are hard, and for captions this information is available for free. We compute the text-text similarity matrix $G_{ij} = \mathbf{t}_i^\top \mathbf{t}_j$ from embeddings the batch already contains, detach it from the computation graph, zero its diagonal, and add it to the logits of Eq. (1):

$$\mathcal{L}_{\text{HN}} = \tfrac{1}{2}\big[\text{CE}\big(s(S + \gamma\bar{G}),\, y\big) + \text{CE}\big(s(S + \gamma\bar{G})^\top,\, y\big)\big], \tag{2}$$

where $\bar{G} = \text{stopgrad}(G) \odot (\mathbf{1} - \mathbb{I})$, with $\odot$ the element-wise product, $\mathbf{1}$ the all-ones matrix, $\mathbb{I}$ the identity (masking the positives), and γ the single hyperparameter (default 0.5).

Interpretation as an adaptive margin. Because $\bar{G}$ is detached, Eq. (2) is exactly Eq. (1) evaluated on shifted logits: negative j of query i competes with an additive handicap γG_{ij} in its favor. Equivalently, the positive must beat every negative by a margin *proportional to how similar that negative's caption is to its own*, a per-pair adaptive generalization of the fixed additive margins used in metric learning. Less similar captions receive smaller shifts; near-duplicate captions ($G_{ij} \to 1$) require larger separation before their loss becomes negligible. The shift is signed: $G_{ij} < 0$ lowers a negative's score. Driving a row loss below any $\varepsilon > 0$ now requires $\Delta_{ij} = S_{ii} - S_{ij} \geq \gamma\bar{G}_{ij} + \frac{1}{s}\log\frac{1}{e^\varepsilon - 1}$ for every $j \neq i$ (Proposition 3). The residual therefore concentrates on exactly the pairs identified in Fig. 1b: with q_{ij} the boosted posterior and $\ell_i^\gamma = -\log q_{ii}$ its row loss, for any two negatives $j, k \neq i$

$$\frac{q_{ij}/q_{ik}}{p_{ij}/p_{ik}} = \exp\big(s\gamma(\bar{G}_{ij} - \bar{G}_{ik})\big) \tag{3}$$

(Proposition 1): an *additive* margin re-weights the gradient *multiplicatively*. At $s \approx 100$ and $\gamma = 0.5$, a caption-similarity gap of 0.1 multiplies the relative gradient weight by $e^5 \approx 148$.

Remark (*Why text–text similarity*)**.** *Hardness could also be estimated from the image-text scores S being optimized, but early in training these are exactly the quantities that are wrong, and reweighting by them makes the supervision a function of the error it should correct. Text–text geometry is instead available before the first gradient step, symmetric across both retrieval directions, and the natural space in which dense-caption difficulty manifests (Fig. 1). The stop-gradient removes the direct differentiable path through $\bar{G}$, so no step is taken toward reshaping the model's own margins; it does not freeze $\bar{G}$ across training, since we recompute it from the current encoder at each step. That residual drift is deliberate: Table 3c shows that recomputing the margins helps, and that freezing them costs up to 7.2 Recall@1.*

Cost. The global boost adds one $B \times B$ matrix product and one masked addition per step; there is no auxiliary forward pass, no extra encoder, no offline extraction. The inference computation is unchanged from the underlying backbone.

2.3 TOKEN-LEVEL ALIGNMENT BRANCH

The boost operates on global embeddings, and composes with supervision at a finer granularity. Following FILIP (Yao et al., 2021), we add a late-interaction term $\mathcal{L}_{\text{tok}}$ that aligns each word token with its most similar image patch (and vice versa), averaged over tokens, with in-batch negatives. The full objective is

$$\mathcal{L} = \mathcal{L}_{\text{HN}} + \lambda\,\mathcal{L}_{\text{tok}}, \qquad \lambda = 1. \tag{4}$$

$\mathcal{L}_{\text{HN}}$ decides *which* pairs the objective spends gradient on; $\mathcal{L}_{\text{tok}}$ decides at *what granularity*. Section 3.5 shows the second question becomes material only once the first is answered, so $\mathcal{L}_{\text{tok}}$ is an optional complement rather than a core component of HN-CLIP. Token matching adds training computation, but both terms act at training time only: no parameters are added and the inference computation is unchanged.

2.4 GRADIENT-DYNAMICS ANALYSIS

An objective that remains unsaturated can supply signal after a less demanding objective becomes small (cf. gradient starvation; Pezeshki et al., 2021). This motivates both auxiliary supervision, as

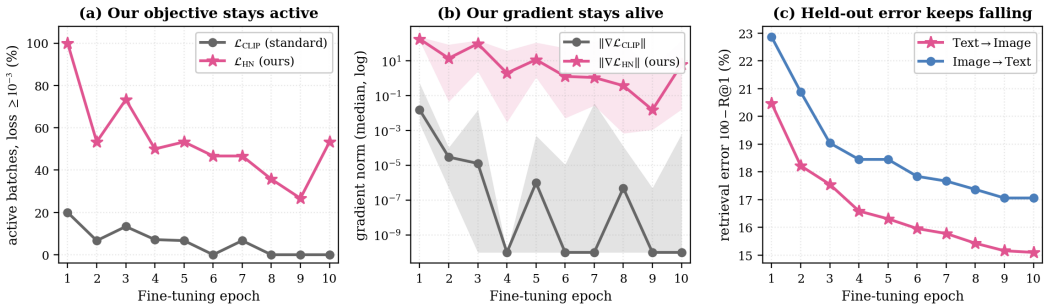

Figure 4: **Empirical gradient analysis** (Long-CLIP-L, DOCCI, 10 epochs; losses and full-parameter gradients measured every 5 optimizer steps, 148 measurements). **(a)** fraction of batches with loss $\geq 10^{-3}$; **(b)** per-epoch gradient-norm medians with interquartile bands, values below the logging resolution plotted at 10^{-10}; **(c)** held-out retrieval error. Quantities are stated in Section 2.4; per-batch spread is discussed in Section C.

in StructXLIP (Ruan et al., 2026), and our boost *without an auxiliary view*. Both regimes can hold on a single query row i: if one confusable pair is still unresolved ($\Delta_{ij} \leq \gamma \bar{G}_{ij}$) while the positive already beats every negative by a gap $d > 0$, its two row losses obey

$$\ell_i^{\gamma} \geq \log 2, \qquad \ell_i^{0} \leq \log\left(1 + (B-1)e^{-sd}\right), \tag{5}$$

the second bound decaying exponentially in sd (Proposition 2). The diagnostics below measure batch losses and full-parameter gradients, which echo this row-level mechanism rather than instantiate the bounds. We run them on a separate 10-epoch diagnostic fine-tuning of Long-CLIP-L on DOCCI (micro-batch 16, accumulation 8). Every 5 optimizer steps we measure both losses on the current batch, the norms of their gradients $\|\nabla_\theta \mathcal{L}_{\text{CLIP}}\|$ and $\|\nabla_\theta \mathcal{L}_{\text{HN}}\|$ w.r.t. all model parameters θ, and the cosine similarity between the two gradients (Fig. 4). The model is updated with $\mathcal{L}_{\text{HN}}$ throughout; $\mathcal{L}_{\text{CLIP}}$ is measured but never optimized, so both objectives are read along a single HN-CLIP trajectory.

Three observations support the design. **(i) Saturation vs. persistence:** within the first epoch, $\mathcal{L}_{\text{CLIP}}$ already falls below 10^{-3} on 80% of measured batches, and from epoch 6 on it does so on essentially all of them along this trajectory: the standard objective simply runs out of work. In contrast, the fraction of batches with $\mathcal{L}_{\text{HN}} \geq 10^{-3}$ never falls below 26% (Fig. 4a). The standard gradient norm falls below the $5{\times}10^{-7}$ logging resolution in 47% of measurements; on the rest, $\mathcal{L}_{\text{HN}}$ exceeds it by 10^2–$10^6\times$ in per-epoch median (Fig. 4b). **(ii) Compatibility:** the gradient cosine is positive in 98.6% of all 148 measurements (μ=0.47, σ=0.36) and in all 78 whose standard gradient exceeds the logging resolution (μ=0.69): the boost steers optimization along directions compatible with the original objective. **(iii) Utility:** persistent signal accompanies improving retrieval: held-out error on DOCCI decreases monotonically from 20.5% to 15.1% (T→I) and 22.9% to 17.1% (I→T) across the same run (Fig. 4c), epochs after $\mathcal{L}_{\text{CLIP}}$ has flattened. The standard objective largely stops providing gradient signal; the boosted one does not. Appendix C repeats this analysis at four training-set scales on two datasets with the same qualitative pattern.

3 Experiments

In this section, we carry out experiments to address the following questions:

- **Q1**: Does HN-CLIP outperform machinery-based fine-tuning methods on dense-caption retrieval? See Section 3.2.
- **Q2**: Does the boost consistently improve existing fine-tuning frameworks in-domain? See Section 3.3.
- **Q3**: Do the sharpened decision boundaries transfer across domains and data scales? See Section 3.4.
- **Q4**: Which component drives the gains, and how sensitive is the single hyperparameter γ? See Section 3.5.

Additional results (full-resolution numbers at deeper ranks (Recall@25/50), seed replication, per-direction convergence, and training-efficiency measurements) can be found in Appendices D and E.

Table 1: **Cross-modal retrieval performance of CLIP-based fine-tuning methods on four dense-caption benchmarks.** We report Recall@K (%) on both *Text→Image* and *Image→Text* settings. All fine-tuned methods start from the same Long-CLIP-L backbone with an identical training budget; Long-CLIP denotes the released checkpoint. Best results in **bold**; second best underlined. Δ denotes the margin over the best competitor per column, with gain in ↑ green.

Method	DOCCI						DCI					
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Long-CLIP[ECCV'24]	78.78	95.24	98.02	66.75	91.92	96.31	67.83	83.19	87.69	64.13	84.84	89.74
FineLIP[CVPR'25]	77.51	96.02	98.41	69.90	93.43	97.45	72.69	87.14	90.65	65.48	86.84	91.00
GOAL[CVPR'25]	81.53	97.02	98.80	80.86	96.24	98.63	77.29	90.25	93.30	74.84	89.94	93.25
StructXLIP[CVPR'26]	84.73	97.69	99.00	82.61	97.08	98.71	75.84	89.94	93.65	74.49	90.05	93.40
HN-CLIP	**88.25**	**98.45**	**99.43**	**86.24**	**98.12**	**99.22**	**80.69**	**92.40**	**95.10**	**78.84**	**91.90**	**94.65**
Δ	↑3.52	↑0.76	↑0.43	↑3.63	↑1.04	↑0.51	↑3.40	↑2.15	↑1.45	↑4.00	↑1.85	↑1.25

Method	Long-DCI						Urban-1K					
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Long-CLIP[ECCV'24]	54.61	72.80	78.33	47.35	73.04	80.10	86.10	96.50	98.10	82.40	96.70	98.30
FineLIP[CVPR'25]	59.24	77.86	83.19	49.52	75.08	82.39	81.50	94.50	97.60	77.90	95.00	98.00
GOAL[CVPR'25]	74.11	92.73	95.75	73.29	92.18	95.78	84.40	96.80	98.60	88.20	97.00	98.60
StructXLIP[CVPR'26]	75.31	93.15	95.92	72.56	92.65	95.75	86.80	97.60	98.90	87.70	96.90	98.30
HN-CLIP	**78.90**	**93.85**	**96.34**	**76.24**	**92.86**	**95.90**	**90.20**	**98.20**	**99.20**	**90.70**	**98.20**	**99.30**
Δ	↑3.59	↑0.70	↑0.42	↑2.95	↑0.21	↑0.12	↑3.40	↑0.60	↑0.30	↑2.50	↑1.20	↑0.70

3.1 Experiment Setup

- **Benchmarks.** DOCCI (Onoe et al., 2024) pairs images with highly discriminative human descriptions; DCI (Urbanek et al., 2024) with dense, mask-aligned captions. Following Choi et al. (2025) we additionally report *Long-DCI*, which evaluates DCI-trained models on a larger caption pool; split sizes and caption lengths are in Table A1. Urban-1K (Zhang et al., 2024) is a 1k-image test-only benchmark of urban scenes; as it provides no training split, models are fine-tuned on Visual Genome paragraph captions (Krause et al., 2017), excluding all Urban-1K evaluation images, and evaluated by transfer.
- **Compared methods.** The main comparison includes the released Long-CLIP (Zhang et al., 2024) checkpoint and FineLIP (Asokan et al., 2025), GOAL (Choi et al., 2025), and StructXLIP (Ruan et al., 2026), the latter three fine-tuned with their official code from the same Long-CLIP-L initialization. Matched plain fine-tuning is isolated in the loss ablations.
- **Implementation.** All methods share the Long-CLIP-L backbone (ViT-L/14; text encoder stretched to 248 tokens), effective batch 128 (micro-batch 16, accumulation 8), a 10-epoch budget and one evaluation protocol. HN-CLIP uses AdamW (Loshchilov & Hutter, 2017) (learning rate 2×10^{-6}, cosine schedule) on Ascend 910B accelerators, and the full objective of Eq. (4) at γ=0.5, λ=1 on all four benchmarks, with no per-benchmark recipe and each term isolated in Table 3b. Baselines otherwise use their official hyperparameters; auxiliary inputs required by GOAL and StructXLIP are generated with their official pipelines. Three-seed replication further preserves HN-CLIP's lead across all four benchmarks (Table A13).

3.2 A1: HN-CLIP Achieves Competitive Performance without Any Machinery

The performance of HN-CLIP surpasses all machinery-based baselines. Table 1 reports the main comparison:

1. HN-CLIP achieves the outright best result in all 24 columns of Table 1 and the best Recall@1 in *all eight* directions, improving over the strongest competitor per column by +2.50–+4.00 Recall@1 (Δ row).
2. The comparison is instructive about *where* the gain comes from: a plain dual encoder that refuses to ignore hard negatives outperforms every machinery-based competitor of Section 4.
3. Consistent with Section 2.4, the largest margins appear at Recall@1, where near-duplicates decide the outcome; at deeper ranks all methods approach ceiling and margins compress, while HN-CLIP remains best across its Long-DCI Recall@5/Recall@10 columns.

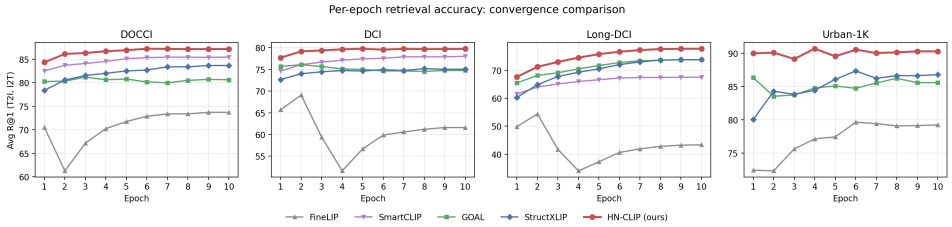

Figure 5: **Convergence comparison.** Average Recall@1 (T→I, I→T) per fine-tuning epoch, with SmartCLIP as an additional reference. Table 1 reports the main evaluation results. On DOCCI, HN-CLIP's first epoch exceeds the FineLIP, GOAL, and StructXLIP results in that table.

Table 2: **Plug-and-play enhancement of our $\mathcal{L}_{\text{HN}}$ on CLIP-based fine-tuning.** Results on DOCCI and Long-DCI for *Text→Image* and *Image→Text* retrieval. Upper: full-parameter fine-tuning; lower: parameter-efficient tuning. Long-CLIP here denotes plain fine-tuning. The boost improves Recall@1 across all variants, and the gains are larger on Long-DCI. Best within each pair in **bold**, with gain in ↑ green.

Method	DOCCI						Long-DCI					
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Long-CLIP[ECCV'24]	86.45	98.00	99.31	84.10	97.84	99.04	70.24	89.62	94.00	68.44	89.35	94.67
+ our $\mathcal{L}_{\text{HN}}$	**88.31**	**98.75**	**99.45**	**86.39**	**98.08**	**99.20**	**78.43**	**92.52**	**95.27**	**76.02**	**91.78**	**94.76**
Δ	↑1.86	↑0.75	↑0.14	↑2.29	↑0.24	↑0.16	↑8.19	↑2.90	↑1.27	↑7.58	↑2.43	↑0.09
FineLIP[CVPR'25]	77.51	96.02	98.41	69.90	93.43	97.45	59.24	77.86	83.19	49.52	75.08	82.39
+ our $\mathcal{L}_{\text{HN}}$	**85.47**	**97.71**	**99.04**	**83.88**	**97.41**	**98.82**	**74.92**	**90.52**	**93.62**	**73.39**	**89.87**	**93.23**
Δ	↑7.96	↑1.69	↑0.63	↑13.98	↑3.98	↑1.37	↑15.68	↑12.66	↑10.43	↑23.87	↑14.79	↑10.84
GOAL[CVPR'25]	81.53	97.02	98.80	80.86	96.24	98.63	74.11	92.73	95.75	73.29	92.18	95.78
+ our $\mathcal{L}_{\text{HN}}$	**86.24**	**98.14**	**99.35**	**84.86**	**97.69**	**99.20**	**84.27**	**94.99**	**96.57**	**82.28**	**94.26**	**96.18**
Δ	↑4.71	↑1.12	↑0.55	↑4.00	↑1.45	↑0.57	↑10.16	↑2.26	↑0.82	↑8.99	↑2.08	↑0.40
StructXLIP[CVPR'26]	84.73	97.69	99.00	82.61	97.08	98.71	75.31	93.15	95.92	72.56	92.65	**95.75**
+ our $\mathcal{L}_{\text{HN}}$	**85.45**	**97.92**	**99.12**	**83.31**	**97.45**	**98.92**	**82.04**	**94.18**	**96.23**	**78.98**	**93.22**	95.65
Δ	↑0.72	↑0.23	↑0.12	↑0.70	↑0.37	↑0.21	↑6.73	↑1.03	↑0.31	↑6.42	↑0.57	↓0.10
LoRA[ICLR'22]	80.49	96.16	98.55	77.96	95.80	98.04	60.08	80.68	86.86	58.97	80.02	86.53
+ our $\mathcal{L}_{\text{HN}}$	**82.82**	**97.14**	**98.86**	**80.84**	**96.49**	**98.63**	**64.87**	**83.09**	**88.35**	**62.79**	**81.77**	**87.31**
Δ	↑2.33	↑0.98	↑0.31	↑2.88	↑0.69	↑0.59	↑4.79	↑2.41	↑1.49	↑3.82	↑1.75	↑0.78
DoRA[ICML'24]	80.76	96.25	98.61	78.20	95.90	98.10	60.76	81.19	87.48	59.65	80.31	87.13
+ our $\mathcal{L}_{\text{HN}}$	**83.41**	**97.33**	**99.00**	**81.41**	**96.65**	**98.67**	**65.46**	**83.57**	**88.61**	**63.38**	**82.15**	**87.56**
Δ	↑2.65	↑1.08	↑0.39	↑3.21	↑0.75	↑0.57	↑4.70	↑2.38	↑1.13	↑3.73	↑1.84	↑0.43

HN-CLIP accelerates the entire training trajectory. Fig. 5 plots per-epoch accuracy, with SmartCLIP as an additional reference. On DOCCI, the first epoch of HN-CLIP (84.3 average Recall@1) already exceeds the FineLIP, GOAL, and StructXLIP results in Table 1; its second epoch (86.1) also exceeds the SmartCLIP reference. Hard-negative supervision does not merely raise the endpoint; it accelerates the entire trajectory.

3.3 A2: The Boost Consistently Improves Existing Frameworks In-Domain

Consistent in-domain plug-and-play improvement. Because $\mathcal{L}_{\text{HN}}$ only modifies the logit matrix of the global contrastive term, it can replace that term inside any fine-tuning framework; we do so in the official code of the six frameworks in Table 2, following the protocol of Ruan et al. (2026) (integration points in Appendix E.1). Table 2 shows consistent in-domain gains: every tested framework improves on both benchmarks, with larger Recall@1 gains on Long-DCI: $+0.70$–$+13.98$ on DOCCI versus $+3.73$–$+23.87$ on Long-DCI, the setting with the largest retrieval gallery (Table A1) and hence the most near-duplicate candidates competing per query. StructXLIP, whose own auxiliary losses already target alignment quality, still benefits, as do the PEFT adapters. The Urban-1K ablation in Appendix E.2 reports the transfer results.

3.4 A3: Sharpened Boundaries Transfer Across Domains and Scales

Across domains. Hard-negative margins could in principle overfit dataset-specific caption statistics; Table A12 shows the opposite. Trained on DCI and transferred to DOCCI, HN-CLIP scores $85.00/83.14$ Recall@1, higher than the FineLIP, GOAL, and StructXLIP *in-domain* DOCCI results,

Table 3: **Ablation studies of HN-CLIP.** Results for **(a)** boost strength γ (Recall@1, λ=1); **(b)** loss components under the same recipe; **(c)** recomputed vs. frozen $\bar{G}$; and **(d)** fixed vs. adaptive margins under the HN-only objective, with the fixed margin matched to the mean adaptive boost at initialization. Results use *Text→Image* and *Image→Text* retrieval. Full Recall@1/5/10 for **(c)** is reported in Table A2. Avg denotes mean Recall@1 over the shown benchmarks and directions. Best per column in **bold**.

(a) boost strength γ

γ	DOCCI R@1	DOCCI R@1	DCI R@1	DCI R@1	Long-DCI R@1	Long-DCI R@1	Urban-1K R@1	Urban-1K R@1	Avg
γ=0	86.45	84.29	79.24	77.54	70.96	68.08	**92.00**	92.90	81.43
γ=0.25	87.82	**86.63**	**80.69**	**79.19**	76.37	74.69	91.70	**93.10**	**83.77**
γ=0.5	**88.25**	86.24	**80.69**	78.84	78.90	76.24	90.20	90.70	83.76
γ=0.75	88.20	85.73	80.64	78.39	79.82	**76.59**	88.80	89.70	83.48
γ=1.0	87.98	85.02	80.04	78.29	**80.05**	76.48	88.20	88.30	83.05

(b) loss components

$\mathcal{L}_{\text{tok}}$	$\mathcal{L}_{\text{HN}}$	DCI R@1	DCI R@5	DCI R@10	Long-DCI R@1	Long-DCI R@5	Long-DCI R@10	Avg
✗	✗	79.39	91.60	94.65	78.34	92.85	95.25	70.24	89.62	94.00	68.44	89.35	94.67	74.10
✓	✗	79.24	91.55	94.25	77.54	**92.95**	**95.35**	70.96	89.52	93.95	68.08	89.24	94.25	73.95
✗	✓	**81.04**	92.20	95.00	78.54	91.95	94.60	78.43	92.52	95.27	76.02	91.78	94.76	78.51
✓	✓	80.69	**92.40**	**95.10**	**78.84**	91.90	94.65	**78.90**	**93.85**	**96.34**	**76.24**	**92.86**	**95.90**	**78.67**

(c) $\bar{G}$ recomputed vs. frozen

$\bar{G}$ source	DOCCI R@1	DOCCI R@1	DCI R@1	DCI R@1	Long-DCI R@1	Long-DCI R@1	Urban-1K R@1	Urban-1K R@1	Avg
Recomputed $\bar{G}$	**88.25**	**86.24**	**80.69**	**78.84**	**78.90**	**76.24**	**90.20**	**90.70**	**83.76**
Frozen $\bar{G}$ (at init)	85.86	82.43	77.79	75.84	77.93	74.75	87.30	83.50	80.68
Δ (recomputed − frozen)	↑2.39	↑3.81	↑2.90	↑3.00	↑0.97	↑1.49	↑2.90	↑7.20	↑3.08

(d) fixed vs. adaptive margins

Margin	DOCCI R@1	DOCCI R@1	DCI R@1	DCI R@1	Long-DCI R@1	Long-DCI R@1	Urban-1K R@1	Urban-1K R@1	Avg
Fixed margin	85.86	82.63	77.54	75.69	76.23	75.31	88.20	83.60	80.63
Adaptive margin (γ=0.5)	**88.31**	**86.39**	**81.04**	**78.54**	**78.43**	**76.02**	**90.60**	**90.20**	**83.69**

and it leads every column of the reverse and of the harder DOCCI→Long-DCI transfer (Tables A8 and A12).

Across data scales. Fig. 6 varies the training fraction on identical subsets. HN-CLIP leads at every fraction, the margin *widens* as data shrinks to 20% (+7.3 Recall@1), and a fifth of the data already beats GOAL trained on all of it. When gradient steps are scarce, spending them on informative negatives matters most. Matched plain fine-tuning results are in Appendix E.2.

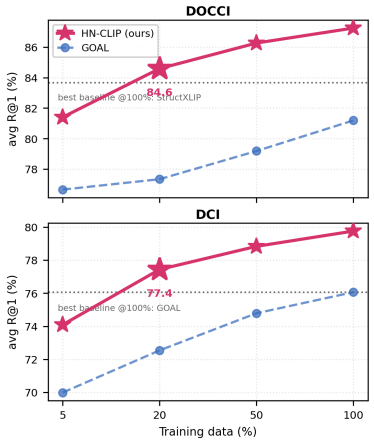

Figure 6: **Sample efficiency** (avg Recall@1, identical subsets). HN-CLIP at 20% of the data clears the best comparator in Table 1 trained on 100% (dotted line); full numbers in Table A10.

3.5 A4: Ablation Studies

Loss components. Table 3b isolates each term under a strictly identical recipe, and the pattern directly tests our diagnosis. Token-level supervision *on its own* leaves average Recall@1 where plain fine-tuning left it, exactly what Section 2.4 predicts: a finer granularity inherits the same uniform treatment of negatives and saturates with it: for this form of token-level supervision, *granularity is not the bottleneck; negative weighting is.* Once $\mathcal{L}_{\text{HN}}$ removes that bottleneck both counts change: alone it gains +1.7/ + 0.2 Recall@1 on DCI and +8.2/ + 7.6 on Long-DCI, and the previously inert token branch adds +0.47/ + 0.22 on Long-DCI, though its DCI effects remain mixed. The boost provides the main gains; token-level supervision adds further improvement on Long-DCI.

Strength of the boost. The γ sweep (Table 3a) shows a broad plateau: every nonzero setting improves in-domain Recall@1 over γ=0, while the nonzero settings differ by only 0.7 average Recall@1. Long-DCI favors larger γ, whereas Urban-1K transfer favors milder boosts. Our default γ=0.5 is within 0.03 Recall@1 of the best setting. Additional temperature and micro-batch analyses are in Appendix B.2; performance remains stable for $B \in \{8, 16, 32\}$ (Table A3).

The drift of $\bar{G}$ is load-bearing. Because the text encoder trains, the recomputed $\bar{G}$ drifts across steps; Table 3c rebuilds the boost from a frozen copy of the initial encoder to test whether that drift helps. The recomputed variant wins the average Recall@1 on all four benchmarks and all 24 columns. Recomputing the geometry lets the margins update as the text encoder trains. The largest frozen-margin penalty is on transfer, where the γ sweep also favors milder boosts (Appendix B.2).

The margin must be allocated, not merely large. A matched fixed-margin control makes this distinction explicit (Table 3d). We replace the adaptive $\gamma\bar{G}_{ij}$ with a single constant margin matched to the mean adaptive boost at initialization for each training source, while keeping the HN-only recipe unchanged. The adaptive margin outperforms the matched fixed margin in all eight retrieval directions across the four evaluation settings, improving average Recall@1 from 80.63 to 83.69. Thus, the gain does not come from imposing a large global margin; it comes from assigning different margins to different negatives. The source ablation in Appendix E.3 further supports this conclusion: random geometry provides no gain, while lexical and image geometry recover the benefit when they carry meaningful pairwise structure.

4 Related Work

Long-text vision-language alignment. Contrastive image-text pre-training (Radford et al., 2021; Jia et al., 2021; Zhai et al., 2023) established the dual-encoder paradigm, whose ∼77-token window fits paragraphs poorly. Long-CLIP (Zhang et al., 2024) relaxes it and is our backbone; parallel work stretches the window (Najdenkoska et al., 2024; Wu et al., 2024) or re-captions the corpus (Fan et al., 2023; Chen et al., 2024). On this backbone, FineLIP (Asokan et al., 2025) inserts a cross-modal module, GOAL (Choi et al., 2025) matches segmented regions, SmartCLIP (Xie et al., 2025) re-weights salient tokens, DreamLIP (Zheng et al., 2024) decomposes captions, StructXLIP (Ruan et al., 2026) aligns edge maps with lexicon-filtered captions, FILIP (Yao et al., 2021) (after ColBERT (Khattab & Zaharia, 2020)) aligns word and patch tokens, and PEFT adapters (Hu et al., 2021; Liu et al., 2024) train fewer weights. All enrich *what* is aligned while inheriting the InfoNCE objective unchanged; HN-CLIP instead repairs *how* negatives within it are weighted, and composes with these frameworks (Section 3.3).

Hard negatives in contrastive learning. The role of negatives in contrastive learning is well established (Oord et al., 2018; He et al., 2020; Chen et al., 2020; Khosla et al., 2020; Wang & Isola, 2020), with prior work exploring importance-based reweighting (Robinson et al., 2020), synthetic mixing (Kalantidis et al., 2020), debiased objectives (Chuang et al., 2020), mined captions for compositionality (Yuksekgonul et al., 2022; Thrush et al., 2022; Hsieh et al., 2023), false-negative relabeling (Li et al., 2023; Byun et al., 2024), hardest-negative mining (Faghri et al., 2017), and score-based up-weighting (Radenovic et al., 2023). These methods largely ask *which negatives to use*; HN-CLIP instead asks *where hardness lives*, reading it off the encoder's pre-trained text–text geometry (Section 2.2). That margin is per-pair and detached, unlike the fixed margins of metric learning (Schroff et al., 2015; Wang et al., 2018; Deng et al., 2019) and their embedding-dependent variants (Kim et al., 2022), and available before the first update, unlike DiHT-style score reweighting, which needs the cross-modal scores it optimizes. To our knowledge, no prior work turns caption-to-caption similarity into adaptive per-negative margins, or identifies this saturation in dense-caption retrieval.

5 Conclusion

Dense-caption retrieval did not need more machinery; it needed an objective that keeps learning. We traced the early saturation of the standard loss to the near-duplicate captions that dominate these datasets, and repaired it with the same text–text geometry that exposed the problem.

AI USE STATEMENT

We have not used generative AI tools for any of the tasks with required disclosure, and the remaining required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for polishing the writing and improving the readability of the manuscript. We have reviewed all AI-assisted work: all AI-assisted edits were checked against the underlying results by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies image–text retrieval using existing benchmarks and pre-trained models. Applications of these models should respect dataset usage terms and privacy, with attention to biases inherited from the underlying data and models.

REPRODUCIBILITY STATEMENT

The hard-negative objective $\mathcal{L}_{\mathrm{HN}}$ is implemented by the 11-line loss in Appendix B; the token-level branch follows FILIP (Yao et al., 2021) and is specified in Section 2.3. All remaining training components follow standard Long-CLIP fine-tuning. All hyperparameters, learning rates for every baseline, and the plug-and-play integration points are specified in Section 3 and Appendix D. All methods use the same training budget and evaluation protocol. Three-seed results together with an additional Long-DCI seed replication verify HN-CLIP's Recall@1 lead (Appendix E). Anonymized code and training scripts are provided in the supplementary material and will be released publicly upon publication.

REFERENCES

Mothilal Asokan, Kebin Wu, and Fatima Albreiki. Finelip: Extending clip's reach via fine-grained alignment with longer text inputs. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14495–14504. IEEE, 2025.

Jaeseok Byun, Dohoon Kim, and Taesup Moon. Mafa: Managing false negatives for vision-language pre-training. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 27304–27314. IEEE, 2024.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. In *European Conference on Computer Vision*, pp. 370–387. Springer, 2024.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PmLR, 2020.

Hyungyu Choi, Young Kyun Jang, and Chanho Eom. Goal: Global-local object alignment learning. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4070–4079. IEEE, 2025.

Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. *Advances in neural information processing systems*, 33:8765–8775, 2020.

Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4690–4699, 2019.

Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. Vse++: Improving visual-semantic embeddings with hard negatives. *arXiv preprint arXiv:1707.05612*, 2017.

Lijie Fan, Dilip Krishnan, Phillip Isola, Dina Katabi, and Yonglong Tian. Improving clip training with language rewrites. *Advances in Neural Information Processing Systems*, 36:35544–35575, 2023.

Roopal Garg, Andrea Burns, Burcu Karagol-Ayan, Yonatan Bitton, Ceslee Montgomery, Yasumasa Onoe, Andrew Bunner, Ranjay Krishna, Jason Michael Baldridge, and Radu Soricut. Imagein-words: Unlocking hyper-detailed image descriptions. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 93–127, 2024.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 9726–9735. IEEE, 2020.

Cheng-Yu Hsieh, Jieyu Zhang, Zixian Ma, Aniruddha Kembhavi, and Ranjay Krishna. Sugarcrepe: Fixing hackable benchmarks for vision-language compositionality. *Advances in neural information processing systems*, 36:31096–31116, 2023.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pp. 4904–4916. PMLR, 2021.

Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. *Advances in neural information processing systems*, 33: 21798–21809, 2020.

Omar Khattab and Matei Zaharia. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pp. 39–48, 2020.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020.

Minchul Kim, Anil K Jain, and Xiaoming Liu. Adaface: Quality adaptive margin for face recognition. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 18729–18738. IEEE, 2022.

Jonathan Krause, Justin Johnson, Ranjay Krishna, and Li Fei-Fei. A hierarchical approach for generating descriptive image paragraphs. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3337–3345. IEEE, 2017.

Haoxuan Li, Yi Bin, Junrong Liao, Yang Yang, and Heng Tao Shen. Your negative may not be true negative: Boosting image-text matching with false negative elimination. In *Proceedings of the 31st ACM international conference on multimedia*, pp. 924–934, 2023.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. Dora: Weight-decomposed low-rank adaptation. *arXiv preprint arXiv:2402.09353*, 2024.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Ivona Najdenkoska, Mohammad Mahdi Derakhshani, Yuki M Asano, Nanne Van Noord, Marcel Worring, and Cees GM Snoek. Tulip: Token-length upgraded clip. *arXiv preprint arXiv:2410.10034*, 2024.

Yasumasa Onoe, Sunayana Rane, Zachary Berger, Yonatan Bitton, Jaemin Cho, Roopal Garg, Alexander Ku, Zarana Parekh, Jordi Pont-Tuset, Garrett Tanzer, et al. Docci: Descriptions of connected and contrasting images. In *European Conference on Computer Vision*, pp. 291–309. Springer, 2024.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Mohammad Pezeshki, Oumar Kaba, Yoshua Bengio, Aaron C Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. *Advances in Neural Information Processing Systems*, 34:1256–1272, 2021.

Filip Radenovic, Abhimanyu Dubey, Abhishek Kadian, Todor Mihaylov, Simon Vandenhende, Yash Patel, Yi Wen, Vignesh Ramanathan, and Dhruv Mahajan. Filtering, distillation, and hard negatives for vision-language pre-training. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6967–6977. IEEE, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. *arXiv preprint arXiv:2010.04592*, 2020.

Zanxi Ruan, Songqun Gao, Qiuyu Kong, Yiming Wang, and Marco Cristani. Structxlip: Enhancing vision-language models with multimodal structural cues. *arXiv preprint arXiv:2602.20089*, 2026.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.

Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5228–5238. IEEE, 2022.

Jack Urbanek, Florian Bordes, Pietro Astolfi, Mary Williamson, Vasu Sharma, and Adriana Romero-Soriano. A picture is worth more than 77 text tokens: Evaluating clip-style models on dense captions. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26690–26699. IEEE, 2024.

Feng Wang and Huaping Liu. Understanding the behaviour of contrastive loss. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2495–2504. IEEE, 2021.

Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 5265–5274. IEEE, 2018.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International conference on machine learning*, pp. 9929–9939. PMLR, 2020.

Wei Wu, Kecheng Zheng, Shuailei Ma, Fan Lu, Yuxin Guo, Yifei Zhang, Wei Chen, Qingpei Guo, Yujun Shen, and Zheng-Jun Zha. Lotlip: Improving language-image pre-training for long text understanding. *Advances in Neural Information Processing Systems*, 37:64996–65019, 2024.

Shaoan Xie, Lingjing Lingjing, Yujia Zheng, Yu Yao, Zeyu Tang, Eric P Xing, Guangyi Chen, and Kun Zhang. Smartclip: Modular vision-language alignment with identification guarantees. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 29780–29790, 2025.

Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. Filip: Fine-grained interactive language-image pre-training. *arXiv preprint arXiv:2111.07783*, 2021.

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.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11941–11952. IEEE, 2023.

Beichen Zhang, Pan Zhang, Xiaoyi Dong, Yuhang Zang, and Jiaqi Wang. Long-clip: Unlocking the long-text capability of clip. In *European conference on computer vision*, pp. 310–325. Springer, 2024.

Kecheng Zheng, Yifei Zhang, Wei Wu, Fan Lu, Shuailei Ma, Xin Jin, Wei Chen, and Yujun Shen. Dreamlip: Language-image pre-training with long captions. In *European Conference on Computer Vision*, pp. 73–90. Springer, 2024.

CONTENTS OF APPENDIX

A Dataset Details **14**

A.1 Benchmarks and splits 14

A.2 Illustrative batches across four settings 14

A.3 What near-duplicates look like 15

A.4 Statistics of the caption geometry 15

B Method Details **21**

B.1 Reference implementation 21

B.2 Theoretical analysis 21

C Extended Gradient Analysis **23**

D Implementation Details and Computational Analysis **24**

D.1 Training setup and backbone choice 24

D.2 Plug-and-play integration 24

D.3 Evaluation protocol 24

E Additional Experimental Analyses **24**

E.1 Plug-and-play, DOCCI and Long-DCI 26

E.2 Full-resolution ablations 26

E.3 Where the boost matrix comes from 26

E.4 Cross-domain evaluation 26

E.5 Robustness to the random seed 26

E.6 Sample efficiency, full numbers 29

E.7 Per-direction convergence 29

E.8 Results at deeper ranks 29

F Additional Qualitative Results **31**

A DATASET DETAILS

A.1 BENCHMARKS AND SPLITS

Table A1 summarizes the four benchmarks, all members of the recent family of hyper-detailed description datasets (Garg et al., 2024). Models are fine-tuned on DOCCI, DCI, or Visual Genome paragraph captions. Long-DCI evaluates DCI-trained models on 7,444 images, versus 1,999 for DCI, using its full-caption annotations with the same 248-token encoder limit, following the protocol of GOAL (Choi et al., 2025). Urban-1K is test-only, so models are fine-tuned on Visual Genome paragraph captions and evaluated by transfer. The last two columns repeat the measurement from Section 2.1: with the pre-trained Long-CLIP-L text encoder, the mean pairwise caption cosine similarity and the mean similarity of each caption's *hardest* companion, the statistics that motivate the per-batch similarities used by $\mathcal{L}_{\text{HN}}$.

A.2 ILLUSTRATIVE BATCHES ACROSS FOUR SETTINGS

Fig. A5 shows the matrix at the heart of HN-CLIP: for one illustrative batch (B=8, same random seed) per setting, the caption-similarity matrix $\bar{G}$ that our loss adds to the logits. The picture is the

Table A1: **Benchmark statistics.** Mean caption length is measured on the evaluation split in words; similarities use the pre-trained Long-CLIP-L text encoder.

Benchmark	#train	#eval	words	pairwise sim	hardest sim
DOCCI	9450	5100	123	0.84	0.93
DCI	5445	1999	133	0.85	0.92
Long-DCI	5445 (=DCI)	7444	134	0.85	0.93
Urban-1K	14579 (VG)	1000	107	0.88	0.94

method's motivation made visible: in these batches the off-diagonal means are 0.82–0.84, so many pairs receive appreciable shifts, each proportional to its caption similarity.

A.3 What near-duplicates look like

Figs. A1 to A4 show representative near-duplicate caption pairs from each evaluation set, retrieved with the *pre-trained* Long-CLIP-L text encoder (queries drawn near the 90th percentile of the hardest-companion distribution, i.e., striking but not extreme). Two different sports cars, two different dirt bikes, two different white horses: the captions describe *different images* yet reach cosine similarities of 0.93–0.97, because long descriptions of natural scenes are compositional recombinations of the same elements (shared content words highlighted). These are exactly the pairs that standard InfoNCE treats as ordinary negatives, and the pairs our margin re-weights hardest.

A.4 Statistics of the caption geometry

Fig. A6 quantifies this across the four evaluation pools with the same encoder: mean pairwise similarity is 0.84–0.88 (A), each caption's hardest companion averages 0.92–0.94 (B), and 86–99% of evaluation-set caption pairs have similarity at least 0.8 (C). At $\gamma{=}0.5$, such a pair would receive a similarity shift of at least 0.4 if sampled together. The evaluation pools also differ in their share of *extreme* ($\bar{G} \geq 0.9$) pairs.



Figure A1: **Near-duplicate caption pairs on DOCCI.** Each row: a test image, its caption, the three nearest *other* captions' images (cosine printed underneath), and the caption texts with shared content words highlighted.



Figure A2: **Near-duplicate caption pairs on DCI.**



Figure A3: **Near-duplicate caption pairs on Long-DCI** (full-length captions).



Figure A4: **Near-duplicate caption pairs on Urban-1K.**

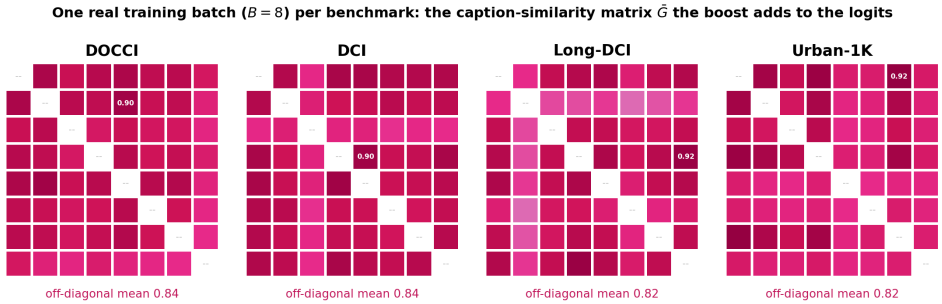

Figure A5: **Illustrative caption batches.** Text–text similarity matrices $\bar{G}$ (pre-trained encoder; diagonal masked, max off-diagonal entry annotated). Dark cells indicate high caption similarity; this matrix, detached and scaled by γ, supplies the global boost. Urban-1K uses its Visual Genome training source; Long-DCI uses its evaluation pool.

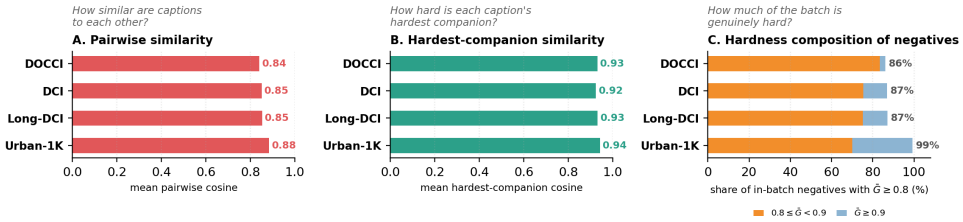

Figure A6: **Statistical analysis of the caption geometry across the four evaluation pools** (pre-trained Long-CLIP-L text encoder). (A) Mean pairwise caption similarity. (B) Mean similarity of each caption's hardest companion. (C) Caption-pair similarity buckets in the evaluation pool, indicating shifts that would apply if sampled together.

B METHOD DETAILS

B.1 REFERENCE IMPLEMENTATION

The complete implementation of $\mathcal{L}_{\text{HN}}$ is shown in Fig. A7, verbatim from our code base; everything else (data loading, optimizer, schedules, evaluation) is standard Long-CLIP fine-tuning.

The complete implementation of $\mathcal{L}_{\text{HN}}$ (PyTorch)

```
def hard_neg_clip_loss(img_n, txt_n, scale, gamma=0.5):
  # img_n, txt_n: L2-normalized [B, D]
  B = img_n.shape[0]
  sim = img_n @ txt_n.t()                    # [B, B]
  txt_sim = (txt_n @ txt_n.t()).detach()
  boost = gamma * txt_sim
  boost.fill_diagonal_(0.)                   # positives
  logits = scale * (sim + boost)
  targets = torch.arange(B, device=sim.device)
  return (F.cross_entropy(logits, targets)
        + F.cross_entropy(logits.t(), targets)) / 2.
```

Figure A7: **Reference implementation.** The complete implementation of $\mathcal{L}_{\text{HN}}$; data loading, optimizer, schedule, and evaluation follow standard Long-CLIP fine-tuning.

B.2 THEORETICAL ANALYSIS

Why does the boost redirect learning toward hard negatives and keep the objective active? We formalize these properties for one query row of Eq. (2). Fix a batch of $B \geq 2$ pairs, the current inverse temperature $s > 0$, and $\gamma \geq 0$. Let $p_{ij} = [\text{softmax}(sS_{i,:})]_j$ and $q_{ij} = [\text{softmax}(s(S_{i,:} + \gamma\bar{G}_{i,:}))]_j$ be the standard and boosted posteriors. Write their row losses as $\ell_i^0 = -\log p_{ii}$ and $\ell_i^\gamma = -\log q_{ii}$. For $j \neq i$, define the unshifted gap $\Delta_{ij} = S_{ii} - S_{ij}$ and the residual adaptive margin $m_{ij} = \Delta_{ij} - \gamma\bar{G}_{ij}$. Since $\bar{G}_{ii} = 0$,

$$\ell_i^\gamma = \log\left(1 + \sum_{j \neq i} \exp(-sm_{ij})\right). \tag{6}$$

All derivatives below are partial derivatives with respect to similarity scores, holding the detached $\bar{G}$ and the current s fixed. The same results apply to the reverse direction after transposing S, since $\bar{G}$ is symmetric; Eq. (2) averages the $2B$ row losses.

Proposition 1 (Relative gradient concentration)**.** *For any two negatives $j, k \neq i$, the boost changes their relative gradient weight by*

$$\frac{(\partial\ell_i^\gamma/\partial S_{ij})/(\partial\ell_i^\gamma/\partial S_{ik})}{(\partial\ell_i^0/\partial S_{ij})/(\partial\ell_i^0/\partial S_{ik})} = \frac{q_{ij}/q_{ik}}{p_{ij}/p_{ik}} = \exp\big(s\gamma(\bar{G}_{ij} - \bar{G}_{ik})\big). \tag{7}$$

Thus, when $\gamma > 0$, a larger caption similarity yields a larger relative weight than under the standard objective.

Proof. The softmax derivatives give $\partial\ell_i^0/\partial S_{ij} = sp_{ij}$ and $\partial\ell_i^\gamma/\partial S_{ij} = sq_{ij}$ for $j \neq i$. Within each posterior ratio the row normalizer cancels, so $p_{ij}/p_{ik} = \exp(s(S_{ij} - S_{ik}))$ and $q_{ij}/q_{ik} = \exp(s(S_{ij} - S_{ik}) + s\gamma(\bar{G}_{ij} - \bar{G}_{ik}))$. Dividing proves the result. □

The effect is multiplicative even though the margin is additive. For example, at $s = 100$ and $\gamma = 0.5$, a caption-similarity gap of 0.1 multiplies the relative gradient weight by $e^5 \approx 148$. This quantifies how caption geometry directs pressure toward confusable pairs.

Proposition 2 (Optimization signal before margin satisfaction)**.** *If some negative $j \neq i$ satisfies $\Delta_{ij} \leq \gamma\bar{G}_{ij}$, then*

$$\ell_i^\gamma \geq \log 2, \qquad \left|\frac{\partial\ell_i^\gamma}{\partial S_{ii}}\right| \geq \frac{s}{2}. \tag{8}$$

Meanwhile, if all unshifted gaps satisfy $\Delta_{ik} \geq d > 0$ for $k \neq i$, the standard objective obeys

$$\ell_i^0 \leq \log(1 + A_d), \qquad \left|\frac{\partial\ell_i^0}{\partial S_{ii}}\right| \leq \frac{sA_d}{1 + A_d}, \quad A_d = (B-1)e^{-sd}. \tag{9}$$

Proof. The first condition gives $m_{ij} \leq 0$, so the corresponding summand in Eq. (6) is at least one. Hence $\ell_i^\gamma \geq \log 2$ and $q_{ii} = (1 + \sum_{k \neq i} e^{-s m_{ik}})^{-1} \leq 1/2$. Using $\partial \ell_i^\gamma / \partial S_{ii} = s(q_{ii} - 1)$ gives Eq. (8). For the standard objective, put $A_i = \sum_{k \neq i} e^{-s\Delta_{ik}}$. The second condition gives $A_i \leq A_d$, while $\ell_i^0 = \log(1 + A_i)$ and $|\partial \ell_i^0 / \partial S_{ii}| = sA_i/(1 + A_i)$. Both expressions increase with A_i, proving Eq. (9). □

The two conditions describe the regime motivating HN-CLIP: the positive can already beat all sampled negatives by a gap d, while a confusable pair still requires the larger separation $\gamma \bar{G}_{ij}$. When $(B-1)e^{-sd}$ is small, standard score gradients are small, yet the boosted objective retains substantial pressure until that adaptive margin is satisfied. This gives an objective-level explanation for the persistent signal examined in Section 2.4.

Proposition 3 (Low loss implies adaptive separation)**.** *For any $\varepsilon > 0$, $\ell_i^\gamma \leq \varepsilon$ implies*

$$\Delta_{ij} \geq \gamma \bar{G}_{ij} + \frac{1}{s} \log \frac{1}{e^\varepsilon - 1} \qquad \textit{for every } j \neq i. \tag{10}$$

Conversely, a sufficient condition for $\ell_i^\gamma \leq \varepsilon$ is

$$\Delta_{ij} \geq \gamma \bar{G}_{ij} + \frac{1}{s} \log \frac{B-1}{e^\varepsilon - 1} \qquad \textit{for every } j \neq i. \tag{11}$$

Proof. By Eq. (6), the loss bound is equivalent to $\sum_{j \neq i} e^{-s m_{ij}} \leq e^\varepsilon - 1$. Each nonnegative summand is then at most $e^\varepsilon - 1$. Taking logarithms and substituting $m_{ij} = \Delta_{ij} - \gamma \bar{G}_{ij}$ proves Eq. (10). Under Eq. (11), each summand is at most $(e^\varepsilon - 1)/(B-1)$; summing over the $B-1$ negatives proves the converse. □

For $\varepsilon < \log 2$, the necessary residual margin is positive: small loss requires separation beyond the caption-dependent shift. The sufficient bound adds $\log(B-1)/s$ to account for the other competitors, and the two bounds coincide for $B = 2$. Together, the three propositions connect caption similarity, gradient allocation, and the separation required to resolve a batch under the boosted objective.

Scope of the stop-gradient, and why the drift is load-bearing. The analysis holds for the detached geometry at each update. We recompute $\bar{G}$ from the *current* text encoder at every step, so although no gradient flows through it, $\bar{G}$ inherits whatever drift the encoder undergoes under the main objective: the margins are stationary within a step but not across training. To determine whether that drift is a defect or a feature, we re-train all three models with $\bar{G}$ built instead from a frozen copy of the pre-trained text encoder, holding seed, schedule, γ, and every other setting fixed (Table A2).

The drift turns out to be load-bearing. The recomputed variant wins all 24 columns, including average Recall@1 on all four benchmarks. The magnitudes illustrate why updating the geometry can matter: $\bar{G}_{ij} = 0.84$ and $\gamma = 0.5$ impose a similarity margin of 0.42, which corresponds to a logit shift of 42 at $s = 100$. Freezing $\bar{G}$ holds that *similarity* margin fixed for all ten epochs; the resulting *logit* shift still varies, since s is learned. Recomputation instead lets a pair's margin change with the text encoder, and it decreases for pairs whose captions grow less similar—an *implicit annealing* of the margin wherever that happens. On Long-DCI, the recomputed variant leads the frozen one by $0.97/1.49$ Recall@1 in T→I / I→T, and by 2.5–3.3 points at Recall@5/Recall@10, benefiting both the top of the ranking and deeper ranks.

This also corroborates the γ sweep from an independent direction. There, milder boosts transferred better and $\gamma \leq 0.25$ was optimal on Urban-1K; here, frozen margins trail the recomputed variant by $2.90/7.20$ Recall@1 on that same transfer benchmark. Two different knobs—the magnitude of the margin and its schedule—therefore agree on where the cost of a never-relaxing margin is largest: on transfer. We therefore keep the recomputed $\bar{G}$ throughout, and note that it is also the cheaper of the two: no second set of weights and no extra forward pass.

Table A2: **Recomputed vs. frozen** $\bar{G}$ (Recall@K, *Text→Image* and *Image→Text*). Building the boost from a frozen copy of the pre-trained text encoder fixes the per-pair similarity margins. The recomputed variant wins all 24 columns. Best per column in **bold**.

	DOCCI						DCI					
$\bar{G}$ source	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Recomputed $\bar{G}$	**88.25**	**98.45**	**99.43**	**86.24**	**98.12**	**99.22**	**80.69**	**92.40**	**95.10**	**78.84**	**91.90**	**94.65**
Frozen $\bar{G}$	85.86	97.61	99.18	82.43	96.82	98.73	77.79	90.80	93.35	75.84	89.49	92.95
Δ (recomputed − frozen)	↑2.39	↑0.84	↑0.25	↑3.81	↑1.30	↑0.49	↑2.90	↑1.60	↑1.75	↑3.00	↑2.41	↑1.70

	Long-DCI						Urban-1K					
$\bar{G}$ source	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Recomputed $\bar{G}$	**78.90**	**93.85**	**96.34**	**76.24**	**92.86**	**95.90**	**90.20**	**98.20**	**99.20**	**90.70**	**98.20**	**99.30**
Frozen $\bar{G}$	77.93	90.84	93.82	74.75	89.54	92.62	87.30	97.70	98.90	83.50	95.90	97.90
Δ (recomputed − frozen)	↑0.97	↑3.01	↑2.52	↑1.49	↑3.32	↑3.28	↑2.90	↑0.50	↑0.30	↑7.20	↑2.30	↑1.40

Interaction with temperature and batch size. Eq. (7) shows that the additional pairwise reweighting factor depends on $s\gamma$, while s also scales the image–text scores S. The sweep in Table 3a characterizes sensitivity to the boost strength.

We additionally vary the micro-batch size $B \in \{8, 16, 32\}$ on DCI while holding the effective optimizer batch fixed at 128 through gradient accumulation of 16, 8, and 4, respectively (Table A3). Average Recall@1 remains stable at 79.62, 79.77, and 80.29, with B=32 performing slightly best. Thus, HN-CLIP is robust to the micro-batch size used to construct S and G, rather than relying on the particular B=16 setting of the main experiments.

Micro-batch B	Accum.	T→I R@1	I→T R@1	**Avg R@1**
8	16	80.54	78.69	79.62
16	8	80.69	78.84	79.77
32	4	**80.94**	**79.64**	**80.29**

Table A3: **Micro-batch sensitivity on DCI.** Effective optimizer batch is fixed at 128; results use the full HN-CLIP objective (γ=0.5, λ=1).

C Extended Gradient Analysis

Reading the median in Fig. 4b. Boosted gradient norms span orders of magnitude across the batches sampled within a single epoch, so the per-epoch median moves with the sampling mix rather than with a trend. The interquartile bands are the more informative statistic: they are disjoint from the standard objective's in nine of the ten epochs, and their upper edge stays above 10^{-3} in gradient norm in every epoch—including the last, where the cosine schedule has decayed the learning rate to 2×10^{-8}, so some batches still induce substantial objective gradients late in training. Standard gradient norms show no such structure: their distribution largely collapses, with the per-epoch median below the logging resolution in five epochs. Panel (a) defines active batches by loss $\geq 10^{-3}$; panel (b) reports gradient norms, which are logged at six decimals, so values below the resulting 5×10^{-7} resolution are plotted at 10^{-10}. Reconstructing those values from the recorded norm ratio gives a median of 4×10^{-9}, some 29 orders of magnitude above the $\sim 10^{-38}$ underflow threshold shared by `fp32` and `bf16`, so the collapse is not an underflow artifact. Both gradients are moreover measured on the same batches, in the same run and the same arithmetic, yet differ by 10^2–$10^6\times$: numerical precision is shared between them and does not explain why only one of them collapses.

Fig. A8 extends the gradient analysis of Fig. 4 across training-data scales: for each fraction of the training set (5/20/50/100%) and each dataset (DOCCI, DCI) we plot the raw traces of (a) both losses, (b) both gradient norms together with their ratio, and (c) the cosine between the two gradients. The qualitative pattern recurs across scales: the standard loss becomes small (gray), while the boosted loss continues to produce signal (pink). Their gradient-norm ratio (dashed), computed on the measurements where the standard gradient is nonzero, stays above one in every panel and grows over training, reaching several orders of magnitude. The gradient direction is also predominantly aligned with the standard objective (median cosine 0.58, positive on 98% of measurements, bottom panels); in a 10^8-dimensional parameter space, where two random directions are near-orthogonal in expectation, this is a strongly compatible update direction. These diagnostics show the same quali-

tative pattern across four training-set scales on DOCCI and DCI: the saturation we diagnose is not an artifact of one training-set size.

D Implementation Details and Computational Analysis

D.1 Training setup and backbone choice

All methods share the Long-CLIP-L backbone (ViT-L/14, positionally stretched 248-token text encoder), micro-batch 16 with $8\times$ gradient accumulation (effective batch 128), and the same 10-epoch training budget. We use a single backbone throughout because the task requires one: captions average 107–134 words on their evaluation splits (Table A1), well beyond the 77-token window of standard CLIP, and Long-CLIP-L is the only publicly released long-context CLIP that every compared method is also built on. Fig. A8 varies training-set size on two datasets, and Table A5 varies the fine-tuning framework. These test data-scale and framework robustness with a shared Long-CLIP-L initialization. HN-CLIP uses AdamW with learning rate 2×10^{-6} and cosine decay; baselines use their official hyperparameters (FineLIP: backbone 10^{-6}, cross-modal module 2×10^{-4}; GOAL and StructXLIP: 5×10^{-6}; LoRA/DoRA adapters: 5×10^{-5}, r=16). Table A4 reports measured training efficiency: HN-CLIP trains $2.4\times$ faster than GOAL and $5.4\times$ faster than StructXLIP on identical hardware while requiring no auxiliary inputs and no offline preprocessing.

Table A4: **Training-efficiency comparison** on DCI fine-tuning (10 epochs, 5.4k images, single Ascend 910B, effective batch 128). Wall-clock for GOAL/StructXLIP is measured from same-device sequential runs minus evaluation overhead; FineLIP ran in parallel and is not attributable. Offline preprocessing (segmentation, edge extraction, LLM filtering) is required by GOAL and StructXLIP but excluded from these figures; including it would widen the gap.

Method	Aux. training inputs	Offline prep.	Wall-clock	Throughput
Long-CLIP (plain FT)	none	none	≈16 min	55.5 img/s
FineLIP	none	none	—	—
GOAL	SAM segments	required	≈41 min	≈22 img/s
StructXLIP	edges + LLM lexicon	required	≈92 min	≈10 img/s
HN-CLIP	**none**	**none**	**17 min**	**53 img/s**

D.2 Plug-and-play integration

For the plug-and-play study we replace the *global* contrastive term of each framework with $\mathcal{L}_{\text{HN}}$ and change nothing else. Long-CLIP: the symmetric InfoNCE over global embeddings is swapped directly. FineLIP: the global term is swapped; the fine-grained cross-modal module and its losses are untouched. GOAL: the original-pair contrastive term is swapped; segment-level alignment terms are untouched. StructXLIP: the RGB–caption global term is swapped; the three structural auxiliary losses are untouched. LoRA/DoRA: adapters (r=16) are trained with $\mathcal{L}_{\text{HN}}$ in place of InfoNCE while the backbone stays frozen. In every case γ=0.5 without per-framework tuning.

D.3 Evaluation protocol

Reported retrieval scores use a shared scoring protocol: we encode the evaluation pool with the released preprocessing (224 center crop, 248-token captions), L2-normalize the embeddings, score by cosine similarity, and report Recall@K in both retrieval directions. The same training budget and evaluation procedure are used for all methods. DCI and Long-DCI share the same training run but differ in gallery size (Table A1). HN-CLIP also retains the highest Long-DCI Recall@1 under the additional seed replication (Table A9).

E Additional Experimental Analyses

Tables A6 to A8 and A10 extend the main-text protocol to *deeper ranks* (Recall@{1, 5, 10, 25, 50}); the remaining tables in this section report Recall@1/5/10. Where a single-run configuration overlaps

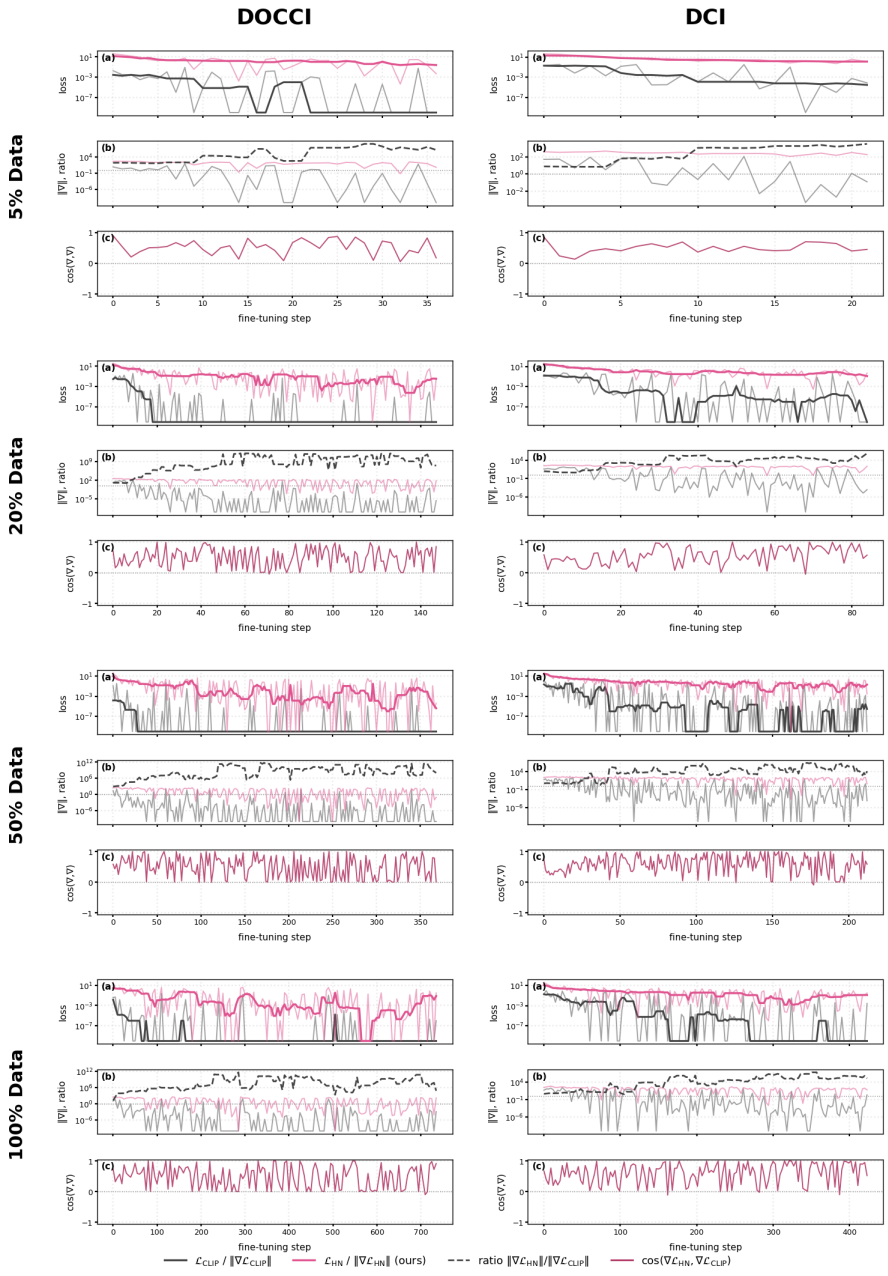

Figure A8: **Gradient dynamics across data scales.** Rows: fraction of the training set; columns: dataset. Per cell: (a) raw loss traces (thin) with running medians (thick), (b) gradient norms and their ratio (dashed), (c) gradient cosine. Log axes in (a,b); values clamped at 10^{-10} for values below the logging resolution.

with the main text, its Recall@1/5/10 values are bit-identical, because it uses the same checkpoint and the same evaluation loop; independent seed replications are reported separately.

E.1 Plug-and-play, DOCCI and Long-DCI

Table A5 reports the plug-and-play study on DOCCI and Long-DCI at full width, in the three-row format (framework / $+\mathcal{L}_{\text{HN}}$ / Δ). Average Recall@1 improves for all twelve framework–dataset combinations, Recall@1 improves in 24 of 24 per-direction cases, with larger gains on Long-DCI (up to $+23.87$ Recall@1 for FineLIP). This establishes consistent in-domain gains across the tested frameworks.

E.2 Full-resolution ablations

Table A6 extends the main-text loss ablation to all four benchmarks and all ranks, and Table A7 does the same for the γ sweep. The full objective has the highest average Recall@1 across the four settings, although HN-only leads on DOCCI and some DCI metrics. On Urban-1K—the only transfer benchmark, where models are fine-tuned on Visual Genome paragraphs and never see urban captions—the unboosted variants lead, in the direction the γ sweep already predicts: milder margins generalize better off the training distribution (Table 3a). At Long-DCI Recall@25/Recall@50, where all four variants lie within 0.8 points of each other, plain fine-tuning is likewise ahead.

E.3 Where the boost matrix comes from

Eq. (2) needs a similarity matrix, not specifically a caption one. We therefore re-train the boosted objective with $\bar{G}$ built from three alternative sources, holding the recipe, $\gamma{=}0.5$, seed, schedule, and epoch budget fixed under the same evaluation protocol: **image**, the batch's image–image geometry; **lexical**, an IDF-weighted bag-of-words cosine over the caption tokens, which uses no encoder at all; and **random**, a fixed random direction per caption, which preserves the machinery but carries no semantics. These runs isolate the global term, so the matched reference is plain fine-tuning and the $\mathcal{L}_{\text{HN}}$-only row rather than the full objective of Eq. (4).

Table A11 reports the result together with the average boost each source applies, measured as the mean off-diagonal $\gamma\bar{G}$ over sampled training batches at $B{=}16$. Random directions in 512 dimensions are near-orthogonal, so that arm applies a mean $|\gamma\bar{G}|$ of 0.02 with no consistent sign—a logit shift below 2, close to a re-run of $\gamma{=}0$—and it leaves accuracy at the plain fine-tuning level: the machinery by itself, with no semantic content in $\bar{G}$, produces no gain. Lexical overlap recovers essentially the whole gain while applying a margin some $14\times$ smaller than the caption geometry (0.03 versus 0.46 in similarity, a logit shift of 3 rather than 46 at $s{\approx}100$): the size of the margin is not what produces the improvement. Image–image geometry performs comparably to the caption geometry, and the caption geometry is the highest of the three on the two-benchmark average while costing nothing extra, since the text embeddings are already computed for the contrastive term. We therefore read the mechanism as requiring a margin allocated by *some* genuine semantic similarity, with caption geometry the cheapest such source rather than a uniquely necessary one.

E.4 Cross-domain evaluation

Table A12 reports the DCI↔DOCCI cross-domain study discussed in Section 3.4: trained on DCI and transferred to DOCCI, HN-CLIP exceeds the FineLIP, GOAL, and StructXLIP *in-domain* DOCCI results, and the reverse transfer keeps HN-CLIP ahead in all six columns. Table A8 evaluates the DOCCI-trained models on Long-DCI, a larger-gallery transfer setting with different caption annotations. HN-CLIP leads every column.

E.5 Robustness to the random seed

Table A13 repeats the main comparison under *three* independent seeds (42, 1337, 3407) and reports mean $\pm$ standard deviation, matching the three-seed reporting convention used by StructXLIP. HN-CLIP's lead in Table 1 is reproduced under replication: it holds the best mean in all 24 columns, including every Recall@1 column on all four benchmarks, by $+2.74$ to $+4.08$. The standard deviations are small—HN-CLIP's Recall@1 standard deviations range from 0.14 to 0.83, remaining

Table A5: **Plug-and-play enhancement of $\mathcal{L}_{\text{HN}}$ on CLIP-based fine-tuning.** Results on DOCCI and Long-DCI for *Text→Image* and *Image→Text* retrieval; per framework we report the official baseline, the same recipe with $\mathcal{L}_{\text{HN}}$ replacing its global InfoNCE term, and the per-column difference Δ (gain in ↑ green, drop in ↓ gray; positive gains of at most 0.15 are shown as ≈0). **Bold** marks the $+\mathcal{L}_{\text{HN}}$ value where it is the better of the pair.

	DOCCI						Long-DCI					
Method	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Long-CLIP	86.45	98.00	99.31	84.10	97.84	99.04	70.24	89.62	94.00	68.44	89.35	94.67
+ our $\mathcal{L}_{\text{HN}}$	**88.31**	**98.75**	**99.45**	**86.39**	**98.08**	**99.20**	**78.43**	**92.52**	**95.27**	**76.02**	**91.78**	**94.76**
Δ	↑1.86	↑0.75	≈0	↑2.29	↑0.24	↑0.16	↑8.19	↑2.90	↑1.27	↑7.58	↑2.43	≈0
FineLIP	77.51	96.02	98.41	69.90	93.43	97.45	59.24	77.86	83.19	49.52	75.08	82.39
+ our $\mathcal{L}_{\text{HN}}$	**85.47**	**97.71**	**99.04**	**83.88**	**97.41**	**98.82**	**74.92**	**90.52**	**93.62**	**73.39**	**89.87**	**93.23**
Δ	↑7.96	↑1.69	↑0.63	↑13.98	↑3.98	↑1.37	↑15.68	↑12.66	↑10.43	↑23.87	↑14.79	↑10.84
GOAL	81.53	97.02	98.80	80.86	96.24	98.63	74.11	92.73	95.75	73.29	92.18	95.78
+ our $\mathcal{L}_{\text{HN}}$	**86.24**	**98.14**	**99.35**	**84.86**	**97.69**	**99.20**	**84.27**	**94.99**	**96.57**	**82.28**	**94.26**	**96.18**
Δ	↑4.71	↑1.12	↑0.55	↑4.00	↑1.45	↑0.57	↑10.16	↑2.26	↑0.82	↑8.99	↑2.08	↑0.40
StructXLIP	84.73	97.69	99.00	82.61	97.08	98.71	75.31	93.15	95.92	72.56	92.65	**95.75**
+ our $\mathcal{L}_{\text{HN}}$	**85.45**	**97.92**	**99.12**	**83.31**	**97.45**	**98.92**	**82.04**	**94.18**	**96.23**	**78.98**	**93.22**	95.65
Δ	↑0.72	↑0.23	≈0	↑0.70	↑0.37	↑0.21	↑6.73	↑1.03	↑0.31	↑6.42	↑0.57	↓0.10
LoRA	80.49	96.16	98.55	77.96	95.80	98.04	60.08	80.68	86.86	58.97	80.02	86.53
+ our $\mathcal{L}_{\text{HN}}$	**82.82**	**97.14**	**98.86**	**80.84**	**96.49**	**98.63**	**64.87**	**83.09**	**88.35**	**62.79**	**81.77**	**87.31**
Δ	↑2.33	↑0.98	↑0.31	↑2.88	↑0.69	↑0.59	↑4.79	↑2.41	↑1.49	↑3.82	↑1.75	↑0.78
DoRA	80.76	96.25	98.61	78.20	95.90	98.10	60.76	81.19	87.48	59.65	80.31	87.13
+ our $\mathcal{L}_{\text{HN}}$	**83.41**	**97.33**	**99.00**	**81.41**	**96.65**	**98.67**	**65.46**	**83.57**	**88.61**	**63.38**	**82.15**	**87.56**
Δ	↑2.65	↑1.08	↑0.39	↑3.21	↑0.75	↑0.57	↑4.70	↑2.38	↑1.13	↑3.73	↑1.84	↑0.43

Table A6: **Loss ablation, full resolution** (all four benchmarks, all ranks). Best per column in **bold**.

		DOCCI										DCI									
$\mathcal{L}_{\text{tok}}$	$\mathcal{L}_{\text{HN}}$	R@1	R@5	R@10	R@25	R@50	R@1	R@5	R@10	R@25	R@50	R@1	R@5	R@10	R@25	R@50	R@1	R@5	R@10	R@25	R@50
✗	✗	86.45	98.00	99.31	**99.86**	**99.98**	84.10	97.84	99.04	99.78	**99.98**	79.39	91.60	94.65	97.25	97.95	78.34	92.85	95.25	97.50	98.50
✓	✗	86.45	97.98	99.43	**99.86**	**99.98**	84.29	97.90	98.94	99.78	99.94	79.24	91.55	94.25	97.00	98.00	77.54	**92.95**	**95.35**	**97.55**	**98.60**
✗	✓	**88.31**	**98.75**	**99.45**	99.84	**99.98**	**86.39**	98.08	99.20	99.82	99.96	**81.04**	92.20	95.00	**97.40**	98.10	78.54	91.95	94.60	97.50	98.20
✓	✓	88.25	98.45	99.43	**99.86**	**99.98**	86.24	**98.12**	**99.22**	**99.84**	99.94	80.69	**92.40**	**95.10**	97.35	**98.15**	**78.84**	91.90	94.65	97.40	98.30

		Long-DCI										Urban-1K									
$\mathcal{L}_{\text{tok}}$	$\mathcal{L}_{\text{HN}}$	R@1	R@5	R@10	R@25	R@50	R@1	R@5	R@10	R@25	R@50	R@1	R@5	R@10	R@25	R@50	R@1	R@5	R@10	R@25	R@50
✗	✗	70.24	89.62	94.00	**97.66**	**98.79**	68.44	89.35	94.67	**98.11**	**98.95**	91.70	**99.10**	**99.50**	**99.80**	99.90	**93.20**	**99.00**	**99.50**	**99.90**	**100.00**
✓	✗	70.96	89.52	93.95	97.29	98.63	68.08	89.24	94.25	97.77	98.94	**92.00**	98.90	99.30	99.70	**100.00**	92.90	98.90	**99.50**	99.80	99.90
✗	✓	78.43	92.52	95.27	97.21	98.47	76.02	91.78	94.76	97.31	98.29	90.60	98.30	99.40	99.60	99.80	90.20	98.30	99.40	99.80	99.80
✓	✓	**78.90**	**93.85**	**96.34**	97.26	98.56	**76.24**	**92.86**	**95.90**	97.42	98.47	90.20	98.20	99.20	99.60	99.80	90.70	98.20	99.30	99.70	99.90

Table A7: γ **sweep, full resolution** (all four benchmarks, all ranks). Best per column in **bold**.

	DOCCI										DCI									
γ	R@1	R@5	R@10	R@25	R@50	R@1	R@5	R@10	R@25	R@50	R@1	R@5	R@10	R@25	R@50	R@1	R@5	R@10	R@25	R@50
γ=0	86.45	97.98	99.43	99.86	**99.98**	84.29	97.90	98.94	99.78	99.94	79.24	91.55	94.25	97.00	98.00	77.54	**92.95**	95.35	97.55	**98.60**
γ=0.25	87.82	98.63	99.45	**99.88**	**99.98**	**86.63**	98.04	99.16	**99.84**	99.96	**80.69**	**92.40**	**95.35**	**97.55**	**98.30**	**79.19**	92.85	**95.40**	**97.70**	98.50
γ=0.5	**88.25**	98.45	99.43	99.86	**99.98**	86.24	**98.12**	99.22	**99.84**	99.94	**80.69**	**92.40**	95.10	97.35	98.15	78.84	91.90	94.65	97.40	98.30
γ=0.75	88.20	**98.67**	99.49	99.82	99.96	85.73	98.00	**99.25**	99.78	**99.98**	80.64	92.20	94.70	97.20	98.25	78.39	91.05	93.90	96.90	98.25
γ=1.0	87.98	98.47	**99.51**	99.82	99.96	85.02	97.69	99.02	99.76	99.96	80.04	92.00	94.40	97.10	**98.30**	78.29	90.80	93.70	96.70	98.00

	Long-DCI										Urban-1K									
γ	R@1	R@5	R@10	R@25	R@50	R@1	R@5	R@10	R@25	R@50	R@1	R@5	R@10	R@25	R@50	R@1	R@5	R@10	R@25	R@50
γ=0	70.96	89.52	93.95	97.29	98.63	68.08	89.24	94.25	97.77	**98.94**	**92.00**	**98.90**	99.30	**99.70**	**100.00**	92.90	**98.90**	**99.50**	**99.80**	**99.90**
γ=0.25	76.37	92.32	95.31	**97.65**	**98.70**	74.69	92.02	95.55	**97.89**	98.72	91.70	98.50	**99.50**	**99.70**	99.80	**93.10**	**98.90**	**99.50**	99.70	99.80
γ=0.5	78.90	**93.85**	**96.34**	97.26	98.56	76.24	**92.86**	**95.90**	97.42	98.47	90.20	98.20	99.20	99.60	99.80	90.70	98.20	99.30	99.70	**99.90**
γ=0.75	79.82	92.42	94.99	97.03	98.13	**76.59**	91.20	94.37	96.79	97.96	88.80	98.50	99.10	99.60	99.60	89.70	97.90	98.90	**99.80**	**99.90**
γ=1.0	**80.05**	92.14	94.64	96.78	97.89	76.48	90.89	93.95	96.47	97.68	88.20	98.20	99.20	99.50	99.60	88.30	97.50	98.60	**99.80**	**99.90**

Table A8: **Cross-domain transfer DOCCI→Long-DCI.** Best per column in **bold**.

Method	DOCCI → Long-DCI									
	R@1	R@5	R@10	R@25	R@50	R@1	R@5	R@10	R@25	R@50
Long-CLIP (zero-shot)	54.61	72.80	78.33	85.29	89.15	47.35	73.04	80.10	86.66	90.60
FineLIP	54.54	72.93	79.42	85.87	89.98	46.44	72.96	80.47	87.04	90.68
GOAL	55.74	74.83	80.63	86.78	90.54	56.60	75.48	81.25	87.45	91.30
StructXLIP	57.70	76.03	81.68	88.06	91.60	57.90	76.36	82.09	87.71	91.03
HN-CLIP	**62.60**	**79.16**	**84.19**	**89.88**	**93.05**	**63.77**	**79.89**	**84.22**	**89.05**	**92.14**

Table A9: **Seed replication on Long-DCI.** $\max|\Delta|$ is the largest absolute difference across the six metrics.

Method	Seed	R@1	R@5	R@10	R@1	R@5	R@10	$\max\|\Delta\|$
HN-CLIP	seed 42	78.90	93.85	96.34	76.24	92.86	95.90	
	seed 43	78.79	92.32	95.12	76.18	91.99	95.15	1.53
GOAL	seed 42	74.11	92.73	95.75	73.29	92.18	95.78	
	seed 43	74.34	92.89	95.77	72.03	92.37	95.69	1.26
StructXLIP	seed 42	75.31	93.15	95.92	72.56	92.65	95.75	
	seed 43	75.58	93.12	96.20	72.39	93.03	95.80	0.38

Table A10: **Sample efficiency, full resolution.** Best per pair in **bold**; Δ is the average-Recall@1 margin of HN-CLIP over GOAL at that fraction.

Data	Method	R@1	R@5	R@10	R@25	R@50	R@1	R@5	R@10	R@25	R@50	Δ
DOCCI												
5%	GOAL	77.88	95.35	97.88	99.59	99.84	75.45	94.65	97.76	99.35	99.84	
	HN-CLIP	**82.73**	**96.94**	**98.69**	**99.69**	**99.84**	**80.08**	**96.22**	**98.35**	**99.57**	**99.98**	↑4.74
20%	GOAL	78.22	95.65	98.22	99.55	99.86	76.47	94.88	97.96	99.39	99.84	
	HN-CLIP	**85.43**	**97.86**	**99.20**	**99.80**	**99.96**	**83.75**	**97.25**	**98.80**	**99.76**	**99.98**	↑7.25
50%	GOAL	80.20	96.47	98.33	99.63	99.88	78.18	95.37	97.96	99.43	99.92	
	HN-CLIP	**87.16**	**98.39**	**99.43**	**99.84**	**99.94**	**85.37**	**97.69**	**99.18**	**99.80**	**99.94**	↑7.07
100%	GOAL	81.53	97.02	98.80	99.78	99.92	80.86	96.24	98.63	99.65	99.90	
	HN-CLIP	**88.25**	**98.45**	**99.43**	**99.86**	**99.98**	**86.24**	**98.12**	**99.22**	**99.84**	**99.94**	↑6.05
DCI												
5%	GOAL	70.94	87.19	90.95	94.95	96.95	69.03	86.89	90.80	94.75	96.85	
	HN-CLIP	**73.59**	**87.39**	**92.30**	**95.10**	**96.95**	**74.59**	**89.19**	**93.20**	**96.00**	**97.15**	↑4.11
20%	GOAL	73.69	87.99	91.40	95.40	97.10	71.39	87.24	91.55	95.45	97.30	
	HN-CLIP	**78.09**	**90.60**	**93.70**	**96.30**	**97.65**	**76.79**	**90.40**	**93.55**	**96.55**	**97.70**	↑4.90
50%	GOAL	75.79	89.19	93.00	95.55	96.80	73.79	88.54	92.65	95.90	97.30	
	HN-CLIP	**79.24**	**91.60**	**94.55**	**97.05**	**97.95**	**78.44**	**91.35**	**94.25**	**96.70**	**98.00**	↑4.05
100%	GOAL	77.29	90.25	93.30	96.10	97.40	74.84	89.94	93.25	96.50	98.15	
	HN-CLIP	**80.69**	**92.40**	**95.10**	**97.35**	**98.15**	**78.84**	**91.90**	**94.65**	**97.40**	**98.30**	↑3.70

Table A11: **Source of the boost matrix** $\bar{G}$ (Recall@K, *Text→Image* and *Image→Text*; $\mathcal{L}_{\text{HN}}$ only, γ=0.5, matched recipe). "Boost" is the mean off-diagonal $\gamma\bar{G}$ over sampled training batches (for random directions the signed mean is $\approx$0 and the mean magnitude is 0.02); caption–caption is the source used by HN-CLIP. Δ is the average-Recall@1 change over plain fine-tuning. Best per column in **bold**.

		DOCCI						
$\bar{G}$ source	Boost	R@1	R@5	R@10	R@1	R@5	R@10	Δ
plain fine-tuning (γ=0)	0	86.45	98.00	99.31	84.10	97.84	99.04	—
random direction	$\approx$0	86.55	98.14	99.41	84.49	97.96	99.12	↑0.24
lexical overlap	0.03	87.78	98.37	99.39	86.37	**98.22**	**99.24**	↑1.80
image–image	0.46	88.04	**98.75**	**99.47**	**86.59**	98.10	99.16	↑2.04
caption–caption	0.46	**88.31**	**98.75**	99.45	86.39	98.08	99.20	↑2.07
		DCI						
plain fine-tuning (γ=0)	0	79.39	91.60	94.65	78.34	**92.85**	95.25	—
random direction	$\approx$0	79.24	91.70	94.80	78.39	92.75	95.50	↓0.05
lexical overlap	0.03	80.54	91.90	**95.40**	**79.19**	92.75	**95.55**	↑1.00
image–image	0.43	80.59	92.05	94.70	79.04	92.00	94.80	↑0.95
caption–caption	0.43	**81.04**	**92.20**	95.00	78.54	91.95	94.60	↑0.92

well below its margins—supporting the Recall@1 lead across these seeds. The Long-DCI Recall@5/Recall@10 columns also favor HN-CLIP in both directions, so the seed-level advantage is not confined to Recall@1.

Table A9 re-trains HN-CLIP, GOAL, and StructXLIP with a second random seed under the identical recipe and evaluates on Long-DCI. For HN-CLIP, the two Recall@1 values differ by only 0.09 on average, and HN-CLIP retains the highest Recall@1 under the second seed (largest Recall@1 deviation 1.26, on GOAL). Across all six metrics, the maximum seed-to-seed difference is 1.53 for HN-CLIP, reflecting modest variation at deeper ranks.

E.6 Sample efficiency, full numbers

Table A10 lists all measurements behind Fig. 6: both datasets, both methods, all fractions and ranks, on identical training subsets. HN-CLIP leads or ties every cell; the average-Recall@1 margin (last column) *widens* as data shrinks to 20% on both datasets, and the 20% rows exceed GOAL's 100% rows on both. The dotted reference line of Fig. 6 is instead the best comparator in Table 1 at 100% (StructXLIP on DOCCI, GOAL on DCI), which the 20% rows also clear. Matched plain fine-tuning results are reported in Table A6.

E.7 Per-direction convergence

Fig. A9 splits Fig. 5 by retrieval direction: HN-CLIP leads in *both* directions in these trajectories from the first epoch, so the averaged curves hide no asymmetry. The split also exposes training instabilities of FineLIP (e.g., an I→T collapse on DCI at epoch 4) that the averaged view smooths over.

E.8 Results at deeper ranks

At Recall@5/Recall@10 the fine-tuned methods approach ceiling on DOCCI and Urban-1K and margins compress there (Table 1); on DCI and Long-DCI the spread stays wide at every rank. On Long-DCI, HN-CLIP is best in all four Recall@5/Recall@10 columns, while continuing to lead all eight Recall@1 columns of Table 1. The second seed preserves its Recall@1 lead with modest variation at deeper ranks. The two mechanisms are complementary rather than competing: Table 2 shows that adding $\mathcal{L}_{\text{HN}}$ *inside* StructXLIP improves it further.

Table A12: **Cross-domain generalization between DCI and DOCCI.** Train on one dataset and test on another. Values are Recall@K (%), using *Text→Image* and *Image→Text* retrieval. In-domain best in ***italic bold***, cross-domain best in **bold**.

Setting	R@1	R@5	R@10	R@1	R@5	R@10
Train on DCI → Test on DCI vs. DOCCI						
Long-CLIP (zero-shot on DCI)	*67.83*	*83.19*	*87.69*	*64.13*	*84.84*	*89.74*
Long-CLIP (zero-shot on DOCCI)	78.78	95.24	98.02	66.75	91.92	96.31
FineLIP (DCI→DCI)	*72.69*	*87.14*	*90.65*	*65.48*	*86.84*	*91.00*
FineLIP (DCI→DOCCI)	80.69	96.47	98.53	65.63	91.51	96.31
GOAL (DCI→DCI)	*77.29*	*90.25*	*93.30*	*74.84*	*89.94*	*93.25*
GOAL (DCI→DOCCI)	80.14	96.25	98.39	77.24	95.22	97.90
StructXLIP (DCI→DCI)	*75.84*	*89.94*	*93.65*	*74.49*	*90.05*	*93.40*
StructXLIP (DCI→DOCCI)	76.94	95.06	97.76	74.24	94.12	97.51
HN-CLIP (DCI→DCI)	***80.69***	***92.40***	***95.10***	***78.84***	***91.90***	***94.65***
HN-CLIP (DCI→DOCCI)	**85.00**	**97.90**	**99.25**	**83.14**	**96.98**	**98.75**
Train on DOCCI → Test on DOCCI vs. DCI						
Long-CLIP (zero-shot on DOCCI)	*78.78*	*95.24*	*98.02*	*66.75*	*91.92*	*96.31*
Long-CLIP (zero-shot on DCI)	67.83	83.19	87.69	64.13	84.84	89.74
FineLIP (DOCCI→DOCCI)	*77.51*	*96.02*	*98.41*	*69.90*	*93.43*	*97.45*
FineLIP (DOCCI→DCI)	68.23	83.84	88.74	63.48	85.79	89.84
GOAL (DOCCI→DOCCI)	*81.53*	*97.02*	*98.80*	*80.86*	*96.24*	*98.63*
GOAL (DOCCI→DCI)	69.58	85.24	89.39	69.88	85.34	89.79
StructXLIP (DOCCI→DOCCI)	*84.73*	*97.69*	*99.00*	*82.61*	*97.08*	*98.71*
StructXLIP (DOCCI→DCI)	69.93	86.74	91.05	71.39	86.99	90.60
HN-CLIP (DOCCI→DOCCI)	***88.25***	***98.45***	***99.43***	***86.24***	***98.12***	***99.22***
HN-CLIP (DOCCI→DCI)	**74.84**	**88.54**	**92.40**	**75.64**	**88.04**	**91.35**

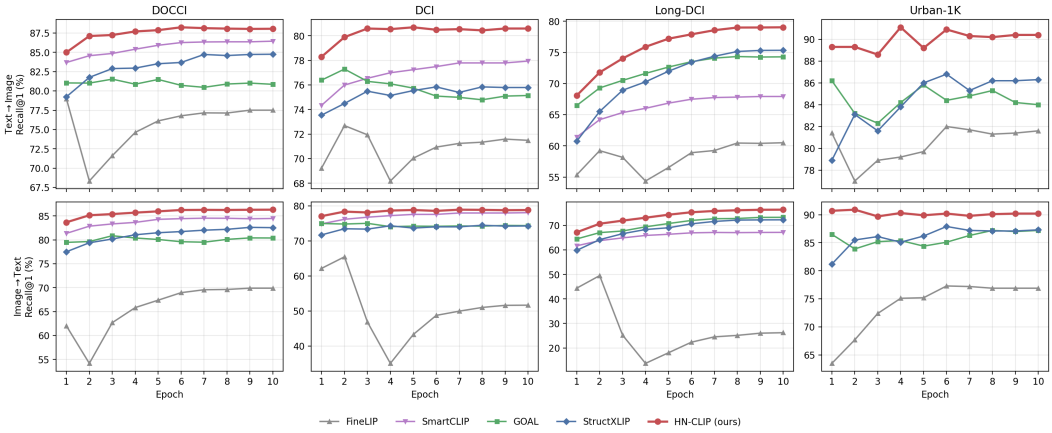

Figure A9: **Per-direction convergence.** Recall@1 per epoch; top: Text→Image, bottom: Image→Text.

Table A13: **Three-seed replication** (seeds 42, 1337, 3407), mean ± std over three independent runs under the same training and evaluation protocol. **Bold** marks the best mean per column, underline the runner-up. Long-CLIP is evaluated zero-shot without fine-tuning and therefore carries no seed variance.

Method	DOCCI T→I R@1	DOCCI T→I R@5	DOCCI T→I R@10	DOCCI I→T R@1	DOCCI I→T R@5	DOCCI I→T R@10
Long-CLIP[ECCV'24] (zero-shot)	78.78	95.24	98.02	66.75	91.92	96.31
FineLIP[CVPR'25]	77.39 ±0.33	95.82 ±0.18	98.36 ±0.05	69.80 ±0.08	93.14 ±0.26	97.37 ±0.09
GOAL[CVPR'25]	81.89 ±0.36	96.97 ±0.04	98.78 ±0.05	80.59 ±0.51	96.19 ±0.10	98.59 ±0.07
StructXLIP[CVPR'26]	84.73 ±0.09	97.64 ±0.05	99.04 ±0.09	82.52 ±0.17	96.97 ±0.09	98.78 ±0.08
HN-CLIP	**88.36** ±0.14	**98.50** ±0.04	**99.48** ±0.05	**86.41** ±0.19	**98.09** ±0.03	**99.24** ±0.02
Δ	↑3.62	↑0.86	↑0.44	↑3.89	↑1.12	↑0.46

Method	DCI T→I R@1	DCI T→I R@5	DCI T→I R@10	DCI I→T R@1	DCI I→T R@5	DCI I→T R@10
Long-CLIP[ECCV'24] (zero-shot)	67.83	83.19	87.69	64.13	84.84	89.74
FineLIP[CVPR'25]	72.77 ±0.21	87.22 ±0.07	90.92 ±0.24	65.55 ±0.35	86.92 ±0.06	90.90 ±0.10
GOAL[CVPR'25]	76.96 ±0.49	90.03 ±0.20	93.23 ±0.26	75.07 ±0.32	89.72 ±0.20	93.32 ±0.06
StructXLIP[CVPR'26]	76.56 ±0.66	90.05 ±0.14	93.52 ±0.12	74.42 ±0.12	90.37 ±0.30	93.68 ±0.26
HN-CLIP	**80.79** ±0.17	**92.22** ±0.16	**94.80** ±0.26	**79.16** ±0.35	**91.93** ±0.15	**94.55** ±0.10
Δ	↑3.83	↑2.17	↑1.28	↑4.08	↑1.57	↑0.87

Method	Long-DCI T→I R@1	Long-DCI T→I R@5	Long-DCI T→I R@10	Long-DCI I→T R@1	Long-DCI I→T R@5	Long-DCI I→T R@10
Long-CLIP[ECCV'24] (zero-shot)	54.61	72.80	78.33	47.35	73.04	80.10
FineLIP[CVPR'25]	59.38 ±0.14	77.85 ±0.08	83.35 ±0.14	49.20 ±0.28	75.10 ±0.10	82.33 ±0.08
GOAL[CVPR'25]	74.58 ±0.50	92.90 ±0.20	95.83 ±0.09	73.40 ±0.46	92.40 ±0.22	95.74 ±0.10
StructXLIP[CVPR'26]	75.51 ±0.21	93.23 ±0.07	96.04 ±0.11	73.05 ±0.50	92.78 ±0.11	95.89 ±0.18
HN-CLIP	**78.84** ±0.15	**93.58** ±0.23	**96.21** ±0.12	**76.14** ±0.30	**92.83** ±0.06	**96.01** ±0.11
Δ	↑3.33	↑0.35	↑0.17	↑2.74	↑0.05	↑0.12

Method	Urban-1K T→I R@1	Urban-1K T→I R@5	Urban-1K T→I R@10	Urban-1K I→T R@1	Urban-1K I→T R@5	Urban-1K I→T R@10
Long-CLIP[ECCV'24] (zero-shot)	86.10	96.50	98.10	82.40	96.70	98.30
FineLIP[CVPR'25]	81.57 ±0.35	94.93 ±0.38	97.23 ±0.33	77.13 ±0.99	94.90 ±0.10	97.50 ±0.44
GOAL[CVPR'25]	86.10 ±1.49	97.00 ±0.53	98.83 ±0.40	88.10 ±0.17	97.10 ±0.36	98.73 ±0.32
StructXLIP[CVPR'26]	85.47 ±1.30	96.67 ±0.86	98.60 ±0.30	87.23 ±0.90	96.57 ±0.42	98.20 ±0.17
HN-CLIP	**89.87** ±0.31	**98.37** ±0.15	**99.33** ±0.12	**91.37** ±0.83	**98.57** ±0.35	**99.27** ±0.06
Δ	↑3.77	↑1.37	↑0.50	↑3.27	↑1.47	↑0.53

F Additional Qualitative Results

Fig. A10 shows Text→Image retrievals on DOCCI queries whose galleries are crowded with near-duplicates, the regime our loss targets. HN-CLIP resolves the discriminative details named in the caption (lettering, flower details, signage), while StructXLIP retrieves appearance-level look-alikes, leaving the ground truth at rank 18–52.

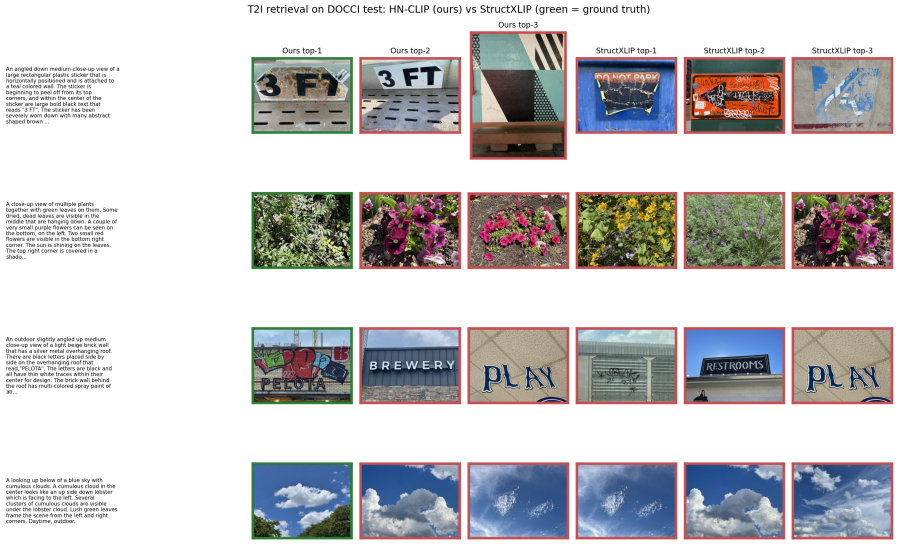

Figure A10: **Qualitative T→I retrieval on DOCCI** (green = ground truth).