TOP-1 IRRETRIEVABILITY OF DENSE RETRIEVAL

Anonymous authors
Paper under double-blind review

ABSTRACT

Dense image retrieval under Maximum Inner Product Search (MIPS) embeds a query and each retrievable item into a common vector space, and returns the item with the largest inner product. Each target is therefore associated with a retrieval cone of query directions from which it is uniquely top-1 ranked. Our work identifies that, when its embedding lies inside the convex hull of the other items' embeddings, the cone is empty, and no query can top-1 retrieve the target. We quantify the extreme rarity of this occurrence in natural data, yet we show such irretrievability can be engineered with near-perfect success rates. When embeddings are normalised, exact irretrievability is impossible. We therefore introduce ϵ-retrievability, relaxing the criterion to a retrieval cone of small volume, and develop methods to make the target difficult to retrieve. We demonstrate these methods on a wide variety of embedding models for both unimodal and cross-modal retrieval tasks often with near-perfect success rate.

1 INTRODUCTION

Dense retrievers embed queries and items with a learned encoder and rank the items by their inner product (Radford et al., 2021; Cherti et al., 2023; Karpukhin et al., 2020). At scale, the best-scoring item is found by Maximum Inner Product Search (MIPS) over a precomputed index (Johnson et al., 2021). Since a query can be any point in the embedding space, one might expect every indexed item to appear as the top result given some query. Recent studies show the limitations of the top-k results a retriever can return (Weller et al., 2025; Wang et al., 2026; Bangachev et al., 2026). We consider a more fundamental question: can every item be top-1 ranked by any query at all?

For each retrievable item, the queries that rank it first form a retrieval cone. This cone is nonempty if and only if the embedding is a vertex of the convex hull of all item embeddings. The geometrical property is closely related to unavailable items in recommender systems (Dean et al., 2020) and classes that a softmax layer can never predict (Demeter et al., 2020; Grivas et al., 2022). In high dimensions, however, almost every embedding is a vertex. For embeddings spread uniformly in a 512-dimensional ball, 10^{77} items are needed before one interior embedding is expected (Appendix C), against roughly 10^{11} images on the public web (Gadre et al., 2023).

We therefore ask whether a chosen target can be *unargmaxified*, that is, made top-1 irretrievable, while the encoder remains fixed and only its inputs change. For unnormalised embeddings (Section 3), we show unargmaxification can be consistently achieved by extending the hull around the target item with inserted unrelated decoys. For constant-norm embeddings (Section 4), excact irretrivability is impossible as every distinct embedding has unit norm and is a vertex with a nonempty retrieval cone. We therefore quantify retrievability through the cone's solid angle and call an item ϵ-retrievable when its normalised cone volume is at least ϵ. We show that each inserted decoy monotonically reduces this retrieval cone, making the target increasingly difficult to recover, until the cone becomes small enough that a gradient-driven query search no longer retrieves the target.

In addition to revealing new insights into the limitation of dense retrieval, our work gives a practical approach to hiding a target image from a trained model. This has application for eliminating unwanted items without the need for re-training, modification of the unwanted item itself, for either privacy or malicious purposes.

These methods are extensively evaluated on nine retrieval models across both unimodal and cross-modal retrieval tasks, consistently demonstrating that a target can be made unargmaxifiable or made no longer ϵ-retrievable with near-perfect success rates.

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.

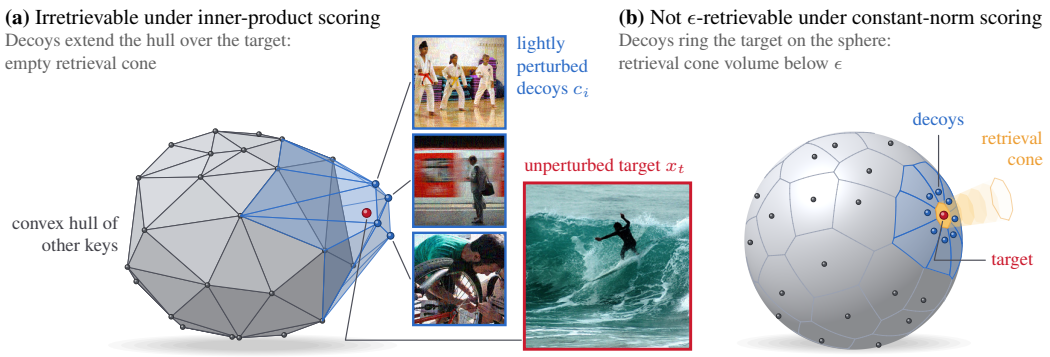

Figure 1: **a)** A target becomes irretrievable by any query when its embedding lies inside the convex hull formed by other embeddings, which can be augmented through decoy insertion of unrelated images. **b)** A target item is no longer ϵ-retrievable when its retrieval cone volume falls below ϵ.

Contributions.

- We provide a novel perspective to the retrievability, geometrically characterising and quantify their success by measuring the top-1 irretrievability and ϵ-retrievability
- We develop a decoy-insertion method that expands the key polytope around an unchanged target and achieves exact, query-independent irretrievability via hull containment.
- For constant-norm embeddings, where exact irretrievability is impossible, we introduce retrieval-cone volume as a measure of retrievability and compute decoys that shrink this cone and prevent query-based recovery.
- We evaluate both methods extensively across nine retrievers and multiple uni/cross-modal tasks.

2 (IR)RETRIEVABILITY FORMULATION

2.1 RETRIEVAL

Retrieval models the similarity between items of a set $\mathcal{X}$, such as images or text documents. Given a predetermined set of keys $\{x_j\}_{j=1}^N \subset \mathcal{X}$ and any query $y \in \mathcal{X}$, our goal is to retrieve the key which is most similar to the query under a similarity function $s : (x, y) \to \mathbb{R}$. In *dense retrieval*, we embed all items via a neural network encoder $f : \mathcal{X} \to \mathbb{R}^d$ which maps the query to its embedding $q = f(y)$ and each item x_j to its embedding $e_j = f(x_j)$, which we write as $E = (e_1, \ldots, e_N)$, $N \geq 2$. We focus on Maximum Inner Product Search (MIPS) where the similarity function is linear: a nonzero query embedding q assigns score $s(q, e_j) = \langle q, e_j \rangle$, with no item-specific bias. The *retrieval cone* for target key t is the set of queries for which it is the unique score maximiser:

$$\mathcal{R}_t(E) = \left\{ q : \langle q, e_t \rangle > \max_{j \neq t} \langle q, e_j \rangle \right\}. \tag{1}$$

This set is convex and invariant under positive rescaling of q. We call a target *irretrievable* when $\mathcal{R}_t(E) = \varnothing$, and the process of making it irretrievable *unargmaxification*. The retrieval cone's normalised solid angle, or its cone volume, is thus

$$\omega_t(E) = \frac{\mathrm{vol}_d(\mathcal{R}_t(E) \cap B_2^d)}{\mathrm{vol}_d(B_2^d)} = \Pr_{q \sim \mathrm{Unif}(\mathbb{S}^{d-1})}[q \in \mathcal{R}_t(E)], \tag{2}$$

where B_2^d and $\mathbb{S}^{d-1}$ are the Euclidean unit ball and sphere. This normalisation is needed because the ordinary volume of a nonempty, full-dimensional cone is infinite.

2.2 GEOMETRY

A polytope is the convex hull of finitely many points. A finite key set therefore gives a V-representation of the *key polytope* $H_E := \mathrm{conv}\{e_1, \ldots, e_N\}$. A key need not be a vertex of H_E.

For target t, let $P_t(E) := \mathrm{conv}\{e_j : j \neq t\}$ denote the convex hull of the keys excluding the target. If $P_t(E) = H_E$ then e_t is not a vertex.

Proposition 1 (Hull criterion and zero cone volume)**.** *Under the scoring model above, for any finite key set E with $N \geq 2$,*

$$e_t \in P_t(E) \quad \Longleftrightarrow \quad \mathcal{R}_t(E) = \varnothing \quad \Longleftrightarrow \quad \omega_t(E) = 0. \tag{3}$$

For pairwise-distinct keys, $\omega_t(E) > 0$ is equivalent to e_t being a vertex of H_E.

By Proposition 1, a target inside $P_t(E)$ cannot be uniquely retrieved by any query. The convex-hull equivalence is the bias-free argmaxability geometry studied for softmax output layers (Demeter et al., 2020; Grivas et al., 2022). The volume equivalence follows because every nonempty retrieval cone is open and therefore has positive solid angle. Appendix B gives a proof.

Under cosine scoring or ℓ_2 normalisation, every key has unit norm. Normalising a query alone leaves its ranking unchanged; normalising the keys changes the key polytope.

Corollary 2 (Constant-norm key sets)**.** *If the keys are pairwise distinct and $\|e_j\|_2 = r > 0$ for every j, then $\mathcal{R}_t(E) \neq \varnothing$ and $\omega_t(E) > 0$ for every target t. Hence exact irretrievability is impossible for a finite, duplicate-free, constant-norm key set.*

For the target-aligned query $q = e_t$, pairwise distinctness and the Cauchy–Schwarz inequality give $\langle q, e_t \rangle = r^2 > \langle q, e_j \rangle$ for every $j \neq t$. Thus $q = e_t \in \mathcal{R}_t(E)$. Appendix B gives a formal proof and discusses the extension to a common strictly convex surface.

3 Irretrievability under inner-product scoring

By Proposition 1, irretrievability under inner-product scoring requires the embedding of an item to be contained in $P_t(E)$. Items with irretrievable embeddings are vanishingly rare: for N embeddings distributed uniformly in a d-dimensional ball, the expected number of interior embeddings reaches one only near $N \approx 2^{d/2}$, which at $d = 512$ is about 10^{77} items (derived in Appendix C), compared with the roughly 10^{11} indexable images on the public web (Gadre et al., 2023).

Indeed, as discussed in Appendix E.1, even in the largest dataset we tested with 1.91 million images, no image is irretrievable by hull containment. Similarly, Appendix E.2 demonstrates that hull containment cannot be achieved through image similarity alone: none of the one million randomly noised copies of a target image falls inside the hull. These motivate the need for a more systematic approach to unargmaxify a target.

We will show that engineered top-1 irretrievability is possible. A target embedding can be placed inside the hull in two ways: directly perturbing the target to move its embedding inside the hull, or perturbing and inserting additional decoys, extending the hull around the target.

3.1 Target-only perturbation

The simplest approach to unargmaxify a target is to perturb the target directly until its embedding lies within the convex hull $P_t(E)$. Writing $\Pi_P(\widehat{e}_t)$ for the Euclidean projection of the perturbed target embedding $\widehat{e}_t$ onto a closed convex set P (Appendix D.2), we define the hull distance $d(\widehat{e}_t, P_t(E)) = \|\widehat{e}_t - \Pi_{P_t(E)}(\widehat{e}_t)\|_2$, and a perturbation is successful when $d(\widehat{e}_t, P_t(E)) = 0$.

Finding such a perturbation is a constrained optimisation problem

$$\min_{\delta} \mathcal{L}(f(x_t + \delta)), \quad \text{s.t.} \quad \|\delta\|_\infty \leq \varepsilon_t,\ x_t + \delta \in [0,1]^D, \tag{4}$$

where ε_t hard-constrains the perturbation magnitude of the target image, enforced by clipping.

In Section 3.3, we investigate several forms of $\mathcal{L}$ that pull the target embedding towards a reference point p_{ref} on the surface of or inside the hull. However, this approach assumes that the reference point p_{ref} is reachable by perturbing the target, which is not guaranteed. Architectures with normalisation blocks, such as LayerNorm, near the output layer may limit the region of embedding space the target can reach, so p_{ref} may not be realisable by any input. This is depicted in Figure 2 and motivates the decoy construction of Section 3.2: instead of perturbing the target embedding to some unreachable region, we construct a convex hull surrounding the target.

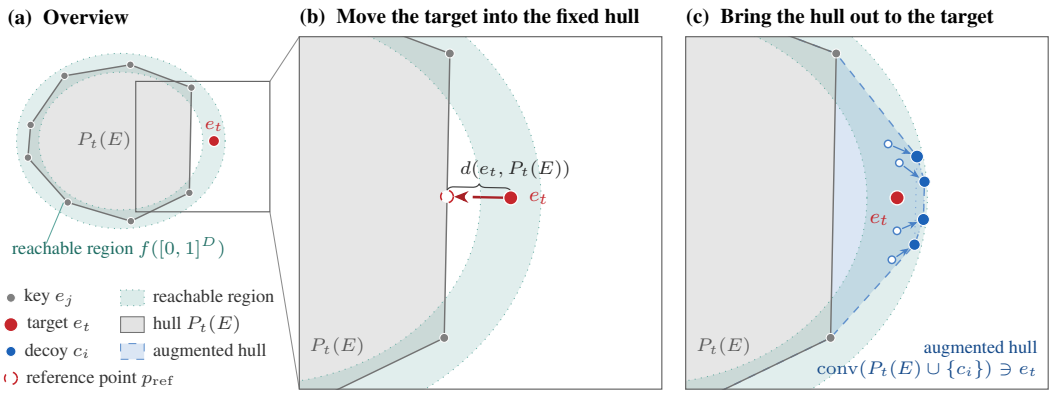

Figure 2: Two routes to irretrievability under inner-product scoring: **b)** perturbing the target into the fixed hull, but p_{ref} may not be reachable. **c)** Decoy construction expands the hull to contain the unperturbed target.

3.2 Robust decoy construction

The second route to unargmaxify the target is to insert decoys, augmenting the hull geometry to place the target inside the hull. This allows the target to remain unperturbed while the hull changes.

Decoy hull construction. Pick a unit direction u whose reverse ray from the target reaches the fixed hull, so that $e_t - R_{\text{in}}u \in P_t(E)$ for some $R_{\text{in}} > 0$. Place the weighted mean of the decoys at

$$\widetilde{c} = e_t + Ru, \qquad e_t = \frac{R_{\text{in}}}{R_{\text{in}} + R}\,\widetilde{c} + \frac{R}{R_{\text{in}} + R}\,(e_t - R_{\text{in}}u), \tag{5}$$

where $R > 0$ is some distance extending outward along u from the target. Then the target e_t lies within the hull augmented by the decoy embeddings $c_1, \ldots, c_C$ whose weighted mean $\sum_i w_i c_i = \widetilde{c}$ for some non-negative weights w_i, where $\sum_i w_i = 1$. Only the weighted mean is fixed, while the individual positions are free.

We choose $u = (e_t - \mu)/\|e_t - \mu\|_2$, pointing from the hull centroid μ to the target. Then let L be the 99th percentile of the key norms $\|e_j\|_2$ over the retrievable item set, and let R_{ref} be the distance from e_t along u to the reference sphere of radius L. We set $R = \phi R_{\text{ref}}$ with a hyperparameter $\phi \in (0, 1)$.

To examine the error tolerance of this construction, let $\widehat{c}_i$ be the realised embedding of decoy i. Its realised weighted mean $\widehat{c} = \sum_i w_i \widehat{c}_i$ misses the desired mean by the aggregate error $\eta = \widehat{c} - \widetilde{c}$. Let A_t be the convex cone of directions whose reverse ray from e_t reaches the hull,

$$A_t = \Big\{ \sum_{j \neq t} \lambda_j (e_t - e_j) \;:\; \lambda_j \geq 0 \Big\}, \tag{6}$$

which contains u by construction. Let θ be the largest angle such that every direction within angle θ of u lies in A_t.

Proposition 3 (Realisation error tolerance)**.** *Assume $e_t \notin P_t(E)$. Then $e_t \in \text{conv}(P_t(E) \cup \{\widehat{c}\})$ if and only if $\widehat{c} - e_t = Ru + \eta \in A_t$. This containment holds for every error with $\|\eta\|_2 \leq R \sin\theta$. It then also implies $e_t \in \text{conv}(P_t(E) \cup \{\widehat{c}_1, \ldots, \widehat{c}_C\})$.*

Appendix B gives the proof. Since the tolerance is proportional to $R = \phi R_{\text{ref}}$, a larger ϕ tolerates a larger realisation error, but also places $\widetilde{c}$ farther from the target, where it may be harder to realise.

Coordinate rescaling. We notice that the convex hull can be extremely thin along certain directions, causing optimisation to fail. We therefore work in a rescaled coordinate system, transformed by a map T defined by tail-rescaled Principal Component Analysis (PCA),

$$z = T(e) := \text{diag}(\kappa)^{-1} V^\top (e - \mu), \qquad \kappa_k = \begin{cases} \bar{\sigma}, & k \leq d - k_*, \\ \sigma_k, & k > d - k_*. \end{cases} \tag{7}$$

The centroid μ, V, and the ordered singular values $\sigma_1 \geq \cdots \geq \sigma_d$ are fitted once on a calibration set. The k_* axes with the smallest singular values are divided by their own standard deviation. k_* is determined by a spectral-knee rule, which is stated in Appendix D.4. Every other axis is divided by the mean $\bar{\sigma} = (d - k_*)^{-1} \sum_{k=1}^{d-k_*} \sigma_k$. Since T is invertible and affine, the convex-hull membership of any embedding is invariant, *i.e.* $e_t \in \mathrm{conv}(S)$ if and only if $T(e_t) \in \mathrm{conv}(T(S))$ for any set S.

Image realisation. We apply the hull construction discussed above in the transformed coordinates, and optimise the decoy images to realise the transformed desired mean $\widetilde{z} = T(\widetilde{c})$. We initialise C decoys from unrelated images ξ_i of the given dataset using greedy Frank–Wolfe (Appendix D.5), such that their weighted embedding mean is close to $\widetilde{z}$. A ridge penalty on $\sum_i w_i^2$ is applied to avoid weight concentration. The weights are held fixed during optimisation to save compute, as recomputing them every step offers very limited performance gain while tripling runtime. All decoy images are optimised jointly:

$$\min_{\{\delta_i\}} \left\| \sum_i w_i T\big(f(Q_8(\xi_i + \delta_i))\big) - \widetilde{z} \right\|_M^2, \quad \text{s.t.} \quad \|\delta_i\|_\infty \leq \varepsilon_c,\ \xi_i + \delta_i \in [0,1]^D, \tag{8}$$

where Q_8 denotes 8-bit quantisation with a straight-through gradient, and ε_c is a hard constraint on the perturbation magnitude of the decoys, enforced by clipping. Soft perturbation constraints, such as adding $\|\delta_i\|_2$ to the objective, empirically lead to a worse perturbation-performance trade-off. $\|v\|_M^2 = v^\top M v$, where the metric M is a damped inverse of the Jacobian Gram matrix, used to account for weaker sensitivity to input in some embedding directions. We estimate the Gram with random pixel probes and periodically recompute M. Without M, gradient descent reduces the error mainly along directions in which the embedding responds strongly to the pixels and stalls the progress. Further details are discussed in Appendix D.6. Finally, we re-encode the decoys and test the hull containment of the target against the whole retrievable set with the full Linear Programme (LP) as specified in Appendix D.1.

3.3 Results

We evaluate the unargmaxification via both target-only perturbation and decoy construction across six retrieval models (OpenCLIP ViT-B/32 (Cherti et al., 2023), OpenAI CLIP RN101 (Radford et al., 2021), SigLIP 2 B/16 (Tschannen et al., 2025), MobileCLIP2 L/14 (Faghri et al., 2025), MetaCLIP 2 H/14 (Chuang et al., 2025), and SigLIP 2 SO400M/16 (Tschannen et al., 2025)) under inner-product scoring. Every method is evaluated on the same 48 Flickr30K (Young et al., 2014) targets against the full $31{,}014$ images as the retrievable items.

Target-only perturbation. We solve Equation (4) using the following three objectives. Write $P = P_t(E)$ for the fixed convex hull and $\widehat{e}_t = f(x_t + \delta)$ for the perturbed target embedding of Section 3.1.

1. **Squared hull distance:** $\mathcal{L}_{\mathrm{sq}}(\widehat{e}_t) = \|\widehat{e}_t - \Pi_P(\widehat{e}_t)\|_2^2$, whose gradient vanishes as $\widehat{e}_t$ approaches the hull.
2. **Unsquared hull distance:** $\mathcal{L}_{\mathrm{unsq}}(\widehat{e}_t) = \|\widehat{e}_t - \Pi_P(\widehat{e}_t)\|_2$, where the gradient is a unit vector.
3. **Hull centroid:** $\mathcal{L}_{\mathrm{centroid}}(\widehat{e}_t) = \|\widehat{e}_t - \mu\|_2^2$, where $\mu = (N-1)^{-1} \sum_{j \neq t} e_j$ is the hull centroid.

The first two objectives update the nearest hull point as $\widehat{e}_t$ moves, whereas the hull centroid is held fixed. The experiments allow $\varepsilon_t = 255/255$ to maximise the performance of the weaker target-only approaches. The Adam optimiser runs for up to $5{,}000$ steps with cosine learning rate decay. Appendix D.3 gives further experimental details.

The experiments demonstrate the limited effectiveness of target-only unargmaxification. Although it can achieve nonzero performance for models with smaller output dimensions (ViT-B/32 and RN101), it fails to unargmaxify targets for models with larger output dimensions (SigLIP 2, MobileCLIP2 and MetaCLIP 2). Empirically, we observe that the pixel gradient $J_f^\top \nabla \mathcal{L}(\widehat{e}_t)$ consistently vanishes, where J_f is the Jacobian of f at $x_t + \delta$.

On the other hand, the target-only perturbation works significantly better on RN101, where the model has no normalisation block in its image architecture. This differs from all other models with normalisation blocks and a linear projection as the last layer, echoing the discussion in Section 3.1.

Table 1: Percentage of targets achieving irretrievability with an empty retrieval cone.

Method	OpenCLIP ViT-B/32 $d = 512$	CLIP RN101 $d = 512$	SigLIP 2 B/16 $d = 768$	MobileCLIP2 L/14 $d = 768$	MetaCLIP 2 H/14 $d = 1024$	SigLIP 2 SO400M $d = 1152$
Squared hull distance	0.0%	22.9%	0.0%	0.0%	0.0%	0.0%
Unsquared hull distance	0.0%	**100.0%**	0.0%	0.0%	0.0%	0.0%
Hull centroid	18.8%	**100.0%**	0.0%	0.0%	0.0%	0.0%
Decoy insertion	**100.0%**	97.9%	**100.0%**	**100.0%**	**95.8%**	**97.9%**

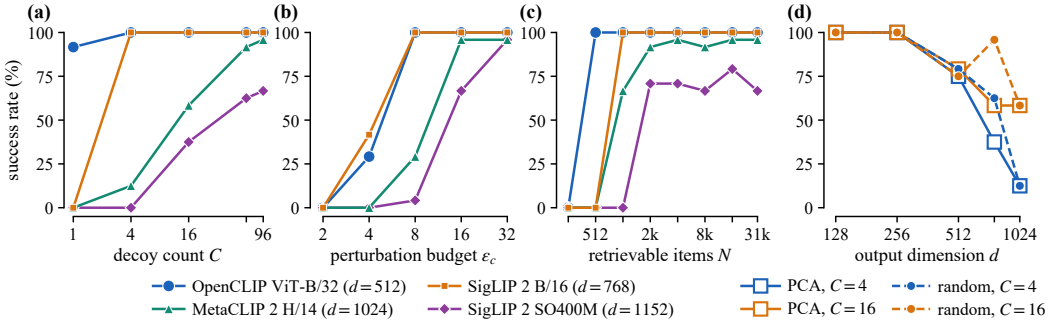

Figure 3: Decoy-based unargmaxification success rate with different **a)** number of decoys C; **b)** decoy perturbation budget ε_c; **c)** number of retrievable items N (excluding the decoys); **d)** MetaCLIP 2 H/14 with its output reduced to d dimensions by keeping the top-d principal components or d randomly chosen features.

Decoy insertion. We freeze the target with $\varepsilon_t = 0$, while optimising 96 decoys with $\varepsilon_c = 16/255$. We set $\phi = 0.15$, except 0.08 for MetaCLIP 2 and 0.25 for SigLIP 2 SO400M. The optimiser runs for at most 3,000 steps, with an early stopping check at 1,500 steps.

We observe in Table 1 that decoy insertion hides the target on all six encoders with near-perfect success rates. Figure 3 further ablates the decoy-based unargmaxification on 24 Flickr30K targets with 1,500 optimisation steps. All other settings follow Table 1. We observe that generally, the higher the number of output dimensions d, the more difficult it is to achieve hull containment, particularly as demonstrated by Figure 3 d). For models with lower output dimensions such as ViT-B/32 and SigLIP 2 B/16, the target can be consistently unargmaxified with as few as $C = 1$–4 decoys, whereas for MetaCLIP 2 H/14 and SigLIP 2 SO400M/16, similar performance can only be achieved with significantly more decoys. Similar conclusions can be drawn with the ε_c and N ablations as well. It can also be observed that the success rate plateaus once N reaches about 2,000. A larger image corpus (Appendix E.4) does not further improve performance either. Appendix E.3 provides further ablation results near the required minimum number of retrievable items to form a nonzero-volume convex hull.

4 Irretrievability under constant-norm scoring

Many retrievers employ inner-product scoring on the normalised keys by using cosine similarity or normalisation at the last layer. For pairwise-distinct normalised keys, Proposition 2 guarantees a nonempty retrieval cone $\mathcal{R}_t(\bar{E})$, where $\bar{e}_j = e_j/\|e_j\|_2$ denotes a normalised key and $\bar{E} = (\bar{e}_1, \ldots, \bar{e}_N)$ denotes the key set. Irretrievability in this case can be characterised by a sufficiently small retrieval cone volume that makes retrieval unlikely. We therefore define the notion:

Definition 1 (ϵ-retrievability)**.** For $\epsilon \geq 0$, target t is *ϵ-retrievable* in E if $\omega_t(E) \geq \epsilon$: a uniformly sampled query direction on $\mathbb{S}^{d-1}$ uniquely retrieves t with probability at least ϵ.

Our objective is thus to minimise the cone volume by inserting decoys around the target in the tangent space of the unit sphere, while leaving the target unchanged.

4.1 DECOY CONSTRUCTION AND OPTIMISATION

Retrieval-cone objective. Let $\mathcal{C} = (c_1, \ldots, c_C)$ be C unit-norm decoy keys and let $\bar{E}^+ = (\bar{e}_1, \ldots, \bar{e}_N, c_1, \ldots, c_C)$ denote the augmented retrievable item set. Each insertion adds one homogeneous half-space

$$\mathcal{R}_t(\bar{E}^+) = \mathcal{R}_t(\bar{E}) \cap \bigcap_{i=1}^{C} \{q : \langle q, \bar{e}_t - c_i \rangle > 0\}. \tag{9}$$

Consequently, decoy insertion monotonically reduces the retrieval cone volume of the target. For a prescribed angular distance $\alpha \in (0, \pi)$, the ideal embedding-space problem is

$$\min_{\mathcal{C}} \ \omega_t(\bar{E}^+) \quad \text{s.t.} \quad c_i \in \mathbb{S}^{d-1}, \qquad \langle c_i, \bar{e}_t \rangle = \cos\alpha. \tag{10}$$

Direct optimisation of the cone volume is impractical as the cone membership is indicator-valued and provides no gradient for backpropagation. We therefore use an angular-separation surrogate that has a closed-form solution and places the decoys around the target.

For $2 \leq C \leq d$, we choose the tangent directions u_i to maximise the smallest angle between any pair, distributing the prescribed decoy keys evenly around the target:

$$\min_{u_1,\ldots,u_C} \ \max_{i \neq i'} \langle u_i, u_{i'} \rangle \quad \text{s.t.} \quad u_i \perp \bar{e}_t, \qquad \|u_i\|_2 = 1. \tag{11}$$

We use the regular-simplex solution, whose directions satisfy $\sum_{i=1}^{C} u_i = 0$. We sample the simplex's orientation and define the corresponding prescribed decoy keys by $c_i^\star = \cos\alpha\, \bar{e}_t + \sin\alpha\, u_i$.

Image realisation and verification. Let $\xi_i \in [0,1]^D$ be the initialised decoy image and define the normalised encoder $\bar{f}(\xi) = f(\xi)/\|f(\xi)\|_2$. With the target and prescribed decoy keys fixed, we optimise the decoy images by solving

$$\min_{\{\delta_i\}} \sum_{i=1}^{C} \left[1 - \langle \bar{f}(Q_8(\xi_i + \delta_i)), c_i^\star \rangle\right] \quad \text{s.t.} \quad \|\delta_i\|_\infty \leq \varepsilon_c, \qquad \xi_i + \delta_i \in [0,1]^D, \tag{12}$$

where Q_8 is 8-bit quantisation with a straight-through gradient, and ε_c is the perturbation budget of the decoys.

4.2 RESULTS

We evaluate query-dependent irretrievability under constant-norm scoring, *e.g.* cosine similarity, or on models whose last layer is a normalisation block. We use the same 48 Flickr30K targets and $31{,}014$ retrievable images as Section 3.3. Text-to-image retrieval uses the same six encoders as in Table 1 and Qwen3-VL-Embedding-2B (Li et al., 2026); image-to-image retrieval uses DINOv2-B/14 (Oquab et al., 2024) and DINOv3-L/16 (Siméoni et al., 2025). Image-to-text and text-to-text retrieval are discussed in Appendix E.6. We keep the target unchanged with $\varepsilon_t = 0$ and optimise $C = 8$ or 64 decoys with target positions at $\alpha = 5°$. Starting from randomly selected initialisation images, we solve Equation (12) with $\varepsilon_c = 16/255$ for $1{,}500$ Adam steps, using an initial learning rate of 0.01 with cosine decay.

Text-to-image retrieval. Two evaluation procedures are considered for text-to-image retrieval: (1) natural caption retrieval, where the queries are the target's human-written captions given in the dataset, and we report the median rather than the mean rank to limit the effect of extreme outliers; (2) text query search, where we adapt ARCA (Jones et al., 2023) to search for a text query that retrieves the target via coordinate descent. Each ARCA search optimises 16 token positions for at most $1{,}200$ coordinate updates. Further details are given in Appendix D.8.

We observe in Table 2, 3 and 4 that, under both evaluation protocols, the decoy images substantially reduce the retrievability of the target across all encoders tested. For caption retrieval in Table 2, the median ranks consistently increase by approximately 30–50% of the inserted decoy count. For searched query retrieval in Table 3, we observe perfect or near-perfect hidden rates across all encoders. These are also consistently observed in Table 4 on a VLM-based embedder (Qwen3-VL), whose last layer is a normalisation block.

Table 2: Median caption rank under decoy insertion.

Decoys C	OpenCLIP ViT-B/32 $d = 512$	CLIP RN101 $d = 512$	SigLIP 2 B/16 $d = 768$	MobileCLIP2 L/14 $d = 768$	MetaCLIP 2 H/14 $d = 1024$	SigLIP 2 SO400M $d = 1152$
0	3	5	2	2	3	1
8	7	7	6	7	8	6
64	**33**	**26.5**	**35.5**	**37.5**	**45.5**	**35**

Table 3: Hidden rate under ARCA text search.

Decoys C	OpenCLIP ViT-B/32 $d = 512$	CLIP RN101 $d = 512$	SigLIP 2 B/16 $d = 768$	MobileCLIP2 L/14 $d = 768$	MetaCLIP 2 H/14 $d = 1024$	SigLIP 2 SO400M $d = 1152$
0	12.5%	43.8%	62.5%	10.4%	8.3%	64.6%
8	66.7%	62.5%	77.1%	27.1%	29.2%	70.8%
64	**100.0%**	**100.0%**	**100.0%**	**100.0%**	**95.8%**	**100.0%**

Further ablation results are presented in Figure 4 with 24 targets on ViT-B/32 except b). From a), we observe that the higher the decoy count C, or the lower the construction angle α, the easier it is to hide the targets. From b), the smaller the cone is, the higher the hidden rate. Additionally, reducing MetaCLIP 2's output dimension by PCA while also reducing C to maintain a fixed cone volume demonstrates that d is a more dominant factor. For the same output dimension in c) and d), the cone volume is fixed by matching the increase in C with a corresponding increase in α. In this case, the higher the α, the lower the hidden rate. Despite a monotonic increase in the caption rank with an increase in C, C is negatively correlated with the hidden rate, indicating that α is the stronger factor.

Image-to-image retrieval. We evaluate DINOv2-B/14 (Oquab et al., 2024) and DINOv3-L/16 (Siméoni et al., 2025) for image-to-image retrieval under the same settings as Table 2. Both models have a normalisation block as their last layer. For image query search, we optimise an image from random noise, unconstrained within the valid pixel range for up to 10,000 Adam steps. Additional query search details are stated in Appendix D.8. Although image query search in the continuous pixel space is generally easier than in the discrete text space, we observe that one can still achieve a perfect hidden rate with sufficiently small α.

Table 4: Caption median rank and text-search hidden rate for Qwen3-VL-Embedding-2B ($d = 2{,}048$).

Decoys C	Caption rank	Hidden rate
0	2	10.4%
8	6	52.1%
64	**27**	**100.0%**

Table 5: Image-to-image hidden rate.

Angle α	DINOv2 B/14 $d = 768$	DINOv3 L/16 $d = 1024$
1°	35.4%	2.1%
0.75°	85.4%	41.7%
0.5°	**100.0%**	**100.0%**

5 Related work

Irretrievability and geometric constraints in MIPS. In sections 3 and 4 we exploit geometric constraints which arise when we use a linear function to score n embeddings in $\mathbb{R}^d$, $n \gg d$. In this setting, only a tiny subset of the full set of rankings is realisable (Cover, 1967; Good & Tideman, 1977). We focus on top-1. From linear programming we know that a target is never ranked as top-1 if the target embedding is interior to the convex hull of the remaining embeddings (Dantzig et al., 1954).

Irretrievability has been studied geometrically under other names depending on the application. In recommender systems, Dean et al. (2020) analyse unavailable items, *i.e.* items that can never be recommended to any user, while Guo et al. (2021) extend this analysis to item combinations. In

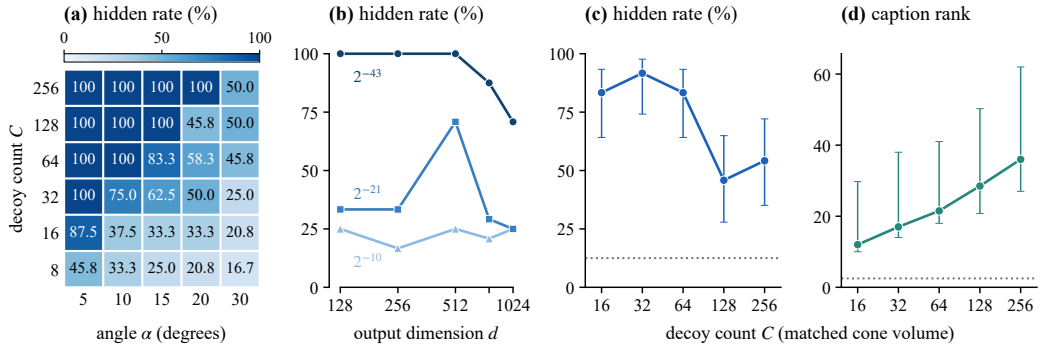

Figure 4: **a)** Hidden rate with different decoy count C and construction angle α; **b)** Hidden rate after PCA reduction to d output dimensions, with decreased C adjusted to maintain a fixed cone volume at $\alpha = 5°$. Line labels denote the cone volume. **c, d)** Hidden rate (c) and median caption rank (d) at the full output dimension with approximately matched prescribed cone volume with increasing C paired with increasing α. The bars show the 95% intervals in (c) and interquartile ranges in (d). Dotted lines mark the no-decoy baseline. The construction parameter details are stated in Appendix D.7.

neural network output layers, previous work has analysed unargmaxable classes (Demeter et al., 2020; Grivas et al., 2022; 2024), *i.e.* classes that can never be produced for any input. Irretrievability is not expected in high dimensions: for retrieval, $2k$ dimensions suffice to realise all top-k rankings (Wang et al., 2026; Weller et al., 2025; Bangachev et al., 2026). Nevertheless, we show that irretrievability can be engineered: we use our geometric understanding to a) learn decoys that make keys irretrievable under MIPS and b) verify irretrievability and quantify ϵ-retrievability.

Decoys for irretrievability. Our decoy constructions are learned via projected first-order optimisation (Madry et al., 2018) and target pretrained feature encoders. They can be thought of as feature collisions which arise in corpus poisoning attacks (Zhong et al., 2023; Li et al., 2025) which target retrieval. A *feature collision* (Shafahi et al., 2018) is an adversarially constructed example that is similar to a chosen item in the input space, but embeds very close to a target embedding. Feature collisions have been used to replace informative top-k results with uninformative ones (Li et al., 2025), attacker-chosen content (Zhou et al., 2020; Zhong et al., 2023; Ben-Tov & Sharif, 2025) or decoys to protect user privacy (Evtimov et al., 2021; Shan et al., 2020). While such approaches work well and may make some targets hard to retrieve, without geometric constraints we cannot guarantee irretrievability and we have no way to quantify how retrievable the targets are for other queries.

On the other hand, our decoys are *geometrically informed feature collisions*: in addition to requiring feature proximity, they enclose the target inside their convex hull. Existing approaches target linear classifiers at fine-tuning time instead of retrieval. Convex Polytope (Zhu et al., 2019) creates decoys by minimising the distance between the target's feature vector and a convex combination of the decoys, subject to an ℓ_∞ budget around the decoy inputs. Bullseye Polytope (Aghakhani et al., 2021) fixes the coefficients to be uniform and minimises the distance to the centroid of the decoys instead, accelerating the attack and placing the target deeper in the polytope. We also keep the coefficients fixed, but choose them via Frank–Wolfe. Moreover, we rescale the coordinates using PCA, without which our decoys failed to make the target irretrievable.

6 Conclusion

Top-1 retrievability under MIPS is a geometric property of the index. An item can be ranked first only if its embedding is a vertex of the convex hull of all item embeddings. Natural high-dimensional indexes almost never contain an item that is not a vertex of the hull, but inserting decoys can make an unperturbed target irretrievable with a near-perfect success rate. Under normalised scoring, where exact irretrievability is impossible, decoys placed around the target shrink its retrieval cone and make the target sufficiently hard to retrieve under query search. Future directions may include extending the existing geometric construction to the top-k irretrievability, as well as over-retrievability, where a target can be perturbed to become much more retrievable against other items.

AI USE STATEMENT

We used generative AI tools to help implement and run the experiments, draft the paper, and provide mathematical insights, under extensive human oversight and manual verification. The authors take full responsibility for the final content. We have not used generative AI tools for literature survey. Polishing was done manually. Synthetic datasets are not applicable for this paper.

REFERENCES

Hojjat Aghakhani, Dongyu Meng, Yu-Xiang Wang, Christopher Kruegel, and Giovanni Vigna. Bullseye polytope: A scalable clean-label poisoning attack with improved transferability. In *2021 IEEE European Symposium on Security and Privacy (EuroS&P)*, pp. 159–178, 2021.

Kiril Bangachev, Guy Bresler, Jonathan Kogan, and Yury Polyanskiy. Is dimensionality a barrier for retrieval models?, 2026. URL https://arxiv.org/abs/2605.23556.

Matan Ben-Tov and Mahmood Sharif. GASLITEing the retrieval: Exploring vulnerabilities in dense embedding-based search. In *ACM CCS*, 2025. URL https://arxiv.org/abs/2412.20953.

Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Yung-Sung Chuang, Yang Li, Dong Wang, Ching-Feng Yeh, Kehan Lyu, Ramya Raghavendra, James Glass, Lifei Huang, Jason Weston, Luke Zettlemoyer, Xinlei Chen, Zhuang Liu, Saining Xie, Wen-tau Yih, Shang-Wen Li, and Hu Xu. Meta CLIP 2: A worldwide scaling recipe. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Thomas M. Cover. The number of linearly inducible orderings of points in d-space. *SIAM Journal on Applied Mathematics*, 15(2):434–439, 1967. URL http://www.jstor.org/stable/2946294.

George B. Dantzig, Alex Orden, and Philip Wolfe. Notes on linear programming: Part I: The generalized simplex method for minimizing a linear form under linear inequality restraints. Research Memorandum RM-1264, RAND Corporation, 1954.

Sarah Dean, Sarah Rich, and Benjamin Recht. Recommendations and user agency: The reachability of collaboratively-filtered information. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, pp. 436–445, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450369367. doi: 10.1145/3351095.3372866. URL https://doi.org/10.1145/3351095.3372866.

David Demeter, Gregory Kimmel, and Doug Downey. Stolen probability: A structural weakness of neural language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 2191–2197, 2020. doi: 10.18653/v1/2020.acl-main.198.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

Ivan Evtimov, Pascal Sturmfels, and Tadayoshi Kohno. FoggySight: A scheme for facial lookup privacy. *Proc. Priv. Enhancing Technol.*, 2021(3):204–226, 2021. doi: 10.2478/POPETS-2021-0044. URL https://doi.org/10.2478/popets-2021-0044.

Fartash Faghri, Pavan Kumar Anasosalu Vasu, Cem Koc, Vaishaal Shankar, Alexander Toshev, Oncel Tuzel, and Hadi Pouransari. MobileCLIP2: Improving multi-modal reinforced training. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=WeF9zolng8.

Marguerite Frank and Philip Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3(1–2), 1956.

Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. DataComp: In search of the next generation of multimodal datasets. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023.

I. J. Good and T. N. Tideman. Stirling numbers and a geometric structure from voting theory. *Journal of Combinatorial Theory, Series A*, 23(1):34–45, 1977. doi: 10.1016/0097-3165(77)90077-2. URL https://www.sciencedirect.com/science/article/pii/0097316577900772.

Andreas Grivas, Nikolay Bogoychev, and Adam Lopez. Low-rank softmax can have unargmaxable classes in theory but rarely in practice. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 6738–6758, 2022. doi: 10.18653/v1/2022.acl-long.465. URL https://aclanthology.org/2022.acl-long.465.

Andreas Grivas, Antonio Vergari, and Adam Lopez. Taming the sigmoid bottleneck: Provably argmaxable sparse multi-label classification. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press, 2024. ISBN 978-1-57735-887-9. doi: 10.1609/aaai.v38i11.29110. URL https://doi.org/10.1609/aaai.v38i11.29110.

Wenshuo Guo, Karl Krauth, Michael Jordan, and Nikhil Garg. The stereotyping problem in collaboratively filtered recommender systems. In *Proceedings of the 1st ACM Conference on Equity and Access in Algorithms, Mechanisms, and Optimization*, EAAMO '21, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450385534. doi: 10.1145/3465416.3483298. URL https://doi.org/10.1145/3465416.3483298.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3), 2021.

Erik Jones, Anca Dragan, Aditi Raghunathan, and Jacob Steinhardt. Automatically auditing large language models via discrete optimization. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, pp. 15307–15329, 2023.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The Open Images Dataset V4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, 128, 2020.

Simon Lacoste-Julien and Martin Jaggi. On the global linear convergence of Frank-Wolfe optimization variants. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

Mingxin Li, Yanzhao Zhang, Dingkun Long, Keqin Chen, Sibo Song, Shuai Bai, Zhibo Yang, Pengjun Xie, An Yang, Dayiheng Liu, Jingren Zhou, and Junyang Lin. Qwen3-VL-Embedding and Qwen3-VL-Reranker: A unified framework for state-of-the-art multimodal retrieval and ranking. *arXiv preprint arXiv:2601.04720*, 2026. URL https://arxiv.org/abs/2601.04720.

Yongkang Li, Panagiotis Eustratiadis, Simon Lupart, and Evangelos Kanoulas. Unsupervised corpus poisoning attacks in continuous space for dense retrieval. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '25, pp. 2452–2462, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400715921. doi: 10.1145/3726302.3730110. URL https://doi.org/10.1145/3726302.3730110.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, 2014.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*, 2018.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, 2021.

Ali Shafahi, W. Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. Poison frogs! Targeted clean-label poisoning attacks on neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Shawn Shan, Emily Wenger, Jiayun Zhang, Huiying Li, Haitao Zheng, and Ben Y. Zhao. Fawkes: Protecting privacy against unauthorized deep learning models. In *Proceedings of the 29th USENIX Conference on Security Symposium*, SEC'20, USA, 2020. USENIX Association. ISBN 978-1-939133-17-5.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. DINOv3. *arXiv preprint arXiv:2508.10104*, 2025. URL https://arxiv.org/abs/2508.10104.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025. URL https://arxiv.org/abs/2502.14786.

Zihao Wang, Hang Yin, Lihui Liu, Hanghang Tong, Yangqiu Song, Ginny Wong, and Simon See. $\mathbb{R}^{2k}$ is theoretically large enough for embedding-based top-k retrieval. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=7Sho2SEu3D.

Orion Weller, Michael Boratko, Iftekhar Naim, and Jinhyuk Lee. On the theoretical limitations of embedding-based retrieval. *CoRR*, abs/2508.21038, 2025. doi: 10.48550/ARXIV.2508.21038. URL https://doi.org/10.48550/arXiv.2508.21038.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2, 2014.

Zexuan Zhong, Ziqing Huang, Alexander Wettig, and Danqi Chen. Poisoning retrieval corpora by injecting adversarial passages. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2023.

Mo Zhou, Zhenxing Niu, Le Wang, Qilin Zhang, and Gang Hua. Adversarial ranking attack and defense. In *European Conference on Computer Vision (ECCV)*, 2020.

Chen Zhu, W. Ronny Huang, Hengduo Li, Gavin Taylor, Christoph Studer, and Tom Goldstein. Transferable clean-label poisoning attacks on deep neural nets. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7614–7623. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/zhu19a.html.

A LIMITATIONS

Our work extends the existing literature with additional geometric insights, comprehensive multimodal evaluation, and detailed ablation studies that examine the irretrievability under different conditions. We identify here the principal limitations.

For irretrievability via hull containment, it is possible for the target to have a large norm near L and lie outside the reference sphere, so the construction in section 3 is not possible without perturbing the target. Although rare, this is the main cause of hiding failures in Table 1.

Since optimisation in discrete token space is much more difficult than in continuous image pixel space, the ARCA text search is of limited capacity. This explains the high no-op baseline ($C = 0$) in Table 3, which can be lowered with stronger optimisation settings, e.g. longer token sequence or more steps, requiring more compute. On the other hand the stronger query search capability can also be countered with stronger hyperparameters on hiding algorithm side, such as larger C and smaller α. The query-dependent results in Section 4.2 are therefore specific to the evaluated search setting.

In our work, we do not explicitly make baseline comparisons against the existing literature, because we identified no directly comparable baselines for the construction in Section 3. Similar works, such as BullsEye (Aghakhani et al., 2021), often require retraining of the embedder, whereas we held the embedder fixed and only perturbed its inputs. For Section 4, the most similar work is FoggySight (Evtimov et al., 2021), even though they considered a different set of evaluation protocols. Their construction is a special case of ours with $\alpha = 0$, and so it is already implicitly part of our evaluation.

B PROOFS AND THEORETICAL DETAILS

Proof of Proposition 1. The first equivalence is known for recommender systems (Dean et al., 2020) and softmax output layers (Demeter et al., 2020; Grivas et al., 2022); we include a proof for completeness. If $e_t = \sum_{j\neq t} \lambda_j e_j$ with $\lambda_j \geq 0$ and $\sum_{j\neq t} \lambda_j = 1$, then $\langle q, e_t\rangle = \sum_{j\neq t} \lambda_j \langle q, e_j\rangle \leq \max_{j\neq t}\langle q, e_j\rangle$ for every q, so $\mathcal{R}_t(E) = \varnothing$. Conversely, if $e_t \notin P_t(E)$, let $p^\star$ be the Euclidean projection of e_t onto $P_t(E)$ and set $q = e_t - p^\star \neq 0$. Since $\langle q, p - p^\star\rangle \leq 0$ for every $p \in P_t(E)$, we have $\langle q, e_t - e_j\rangle \geq \langle q, e_t - p^\star\rangle = \|q\|_2^2 > 0$ for every $j \neq t$, so $q \in \mathcal{R}_t(E)$.

For the second equivalence, $\mathcal{R}_t(E)$ is defined by finitely many strict linear inequalities and is therefore open. A nonempty open cone meets B_2^d in a set of positive volume, so $\mathcal{R}_t(E) \neq \varnothing$ implies $\omega_t(E) > 0$; the converse is immediate from Equation (2). □

Proof of Proposition 2. We prove the more general statement that every key is retrievable when the keys are pairwise distinct and lie on the boundary of a common strictly convex body K, *i.e.* a compact convex set with nonempty interior whose boundary contains no line segment. The corollary is the case $K = rB_2^d$, whose outward normal at e_t is e_t itself. Let $q \neq 0$ be an outward normal of K at e_t, so that $\langle q, x\rangle \leq \langle q, e_t\rangle$ for every $x \in K$. By strict convexity, equality holds only at $x = e_t$, so $\langle q, e_j\rangle < \langle q, e_t\rangle$ for every $j \neq t$. Hence $q \in \mathcal{R}_t(E)$, and $\omega_t(E) > 0$ by Proposition 1. □

Proof of Proposition 3. Let $v = \widehat{c} - e_t = Ru + \eta$. We first show that $e_t \in \mathrm{conv}(P_t(E) \cup \{\widehat{c}\})$ if and only if $v \in A_t$. Normalising the coefficients in Equation (6) shows that the nonzero elements of A_t are exactly the vectors $\lambda(e_t - p)$ with $\lambda > 0$ and $p \in P_t(E)$. If $v = \lambda(e_t - p)$ is such a vector, then $e_t = \frac{1}{1+\lambda}\,\widehat{c} + \frac{\lambda}{1+\lambda}\,p$, and if $v = 0$, then $e_t = \widehat{c}$; either way $e_t \in \mathrm{conv}(P_t(E) \cup \{\widehat{c}\})$. Conversely, if $e_t = \beta\widehat{c} + (1-\beta)p$ with $\beta \in [0, 1]$ and $p \in P_t(E)$, then $\beta > 0$ because $e_t \notin P_t(E)$, and rearranging gives $v = \frac{1-\beta}{\beta}(e_t - p) \in A_t$.

The admissible errors therefore form the translated cone $A_t - Ru$, which contains every η with $\|\eta\|_2 \leq r$ exactly when A_t contains the ball of radius r about Ru. For $r < R$, the rays that meet this ball are those within angle $\arcsin(r/R)$ of u, so a cone contains the ball exactly when it contains every unit vector within that angle of u. Taking θ to be the largest angle such that every unit vector within angle θ of u lies in A_t, the largest admissible radius is $R\sin\theta$. This uses $\theta < \pi/2$, which holds because a hyperplane strictly separating e_t from $P_t(E)$ keeps every nonzero element of A_t strictly on one side. Finally, $\widehat{c}$ is a convex combination of $\widehat{c}_1, \ldots, \widehat{c}_C$, so containment in $\mathrm{conv}(P_t(E) \cup \{\widehat{c}\})$ implies containment in $\mathrm{conv}(P_t(E) \cup \{\widehat{c}_1, \ldots, \widehat{c}_C\})$. □

The witness bound $\sin\theta \geq \rho/R_{\text{in}}$ stated after Proposition 3 follows in the same way. If the ball of radius ρ about $p_{\text{w}} = e_t - R_{\text{in}}u$ lies in $P_t(E)$, then $e_t - p = R_{\text{in}}u + (p_{\text{w}} - p) \in A_t$ for every p in it, so A_t contains the ball of radius ρ about $R_{\text{in}}u$. As $\rho < R_{\text{in}}$ (otherwise the ball would contain e_t), every unit vector within angle $\arcsin(\rho/R_{\text{in}})$ of u lies in A_t.

C Expected number of images for non-optimised hull containment

We estimate how many keys are needed before at least one is expected to lie inside the convex hull of the other embeddings, and hence to be irretrievable by Proposition 1.

Upper bound. Consider $e_1, \ldots, e_N$ be independent and uniform in the unit ball B_2^d. For $e_t \in P_t(E)$, $\langle e_t, e_j\rangle \geq \|e_t\|_2^2$ for some $j \neq t$. We can rewrite it as $\|e_t - e_j/2\|_2 \leq \|e_j\|_2/2$, meaning the key e_t lies in the ball whose diameter is the segment from 0 to e_j. This ball is contained in B_2^d and occupies a fraction $(\|e_j\|_2/2)^d$ of its volume, and $\|e_j\|_2$ has density $d\,r^{d-1}$ within $[0,1]$, so $\mathbb{E}\|e_j\|_2^d = \int_0^1 r^d\, d\, r^{d-1}\,\mathrm{d}r = 1/2$. Hence

$$\Pr\big[\langle e_t, e_j\rangle \geq \|e_t\|_2^2\big] = 2^{-d}\,\mathbb{E}\|e_j\|_2^d = 2^{-(d+1)}. \tag{13}$$

Let I be the number of interior keys, a union bound over the $N(N-1)$ ordered pairs therefore gives

$$\mathbb{E}[I] \leq N(N-1)\,2^{-(d+1)} < N^2\,2^{-(d+1)}, \tag{14}$$

which is below one for all $N \leq 2^{(d+1)/2}$.

Since $\Pr[\|e_j\|_2 \leq \rho] = \rho^d$, the shortest of the N keys has norm about $N^{-1/d}$. Along a fixed unit direction u, $\Pr[\langle u, e_j\rangle \geq \tau] \approx (1-\tau^2)^{d/2}$, so the largest of the N projections onto u is about $\tau_N = (1 - N^{-2/d})^{1/2}$. A union bound over a fine net of directions extends this to all u simultaneously, so the hull of the remaining keys contains the centred ball of radius about τ_N. The shortest key is therefore interior when $N^{-1/d} \leq \sqrt{1 - N^{-2/d}}$. The two arguments agree: the first interior key is expect to exist at $N^\star \approx 2^{d/2}$.

At $d = 512$, the output dimension of ViT-B/32, this gives $N^\star \approx 2^{256} \approx 10^{77}$. Real embeddings are not uniform in a ball, so these numbers indicate scale rather than exact counts, and Appendix E.1 tests real datasets directly.

D Experimental and implementation details

D.1 Irretrievability verification

We verify irretrievability through hull containment by checking whether $\mathcal{R}_t(E)$ is empty with the Chebyshev-centre Linear Programme (LP) used in Grivas et al. (2022), with a fixed query bound $B > 0$:

$$\begin{aligned} \max_{q, r_{\text{LP}}} \quad & r_{\text{LP}} \\ \text{s.t.} \quad & (e_j - e_t)^\top q + \|e_j - e_t\|_2 r_{\text{LP}} \leq 0 \quad (j \neq t), \\ & -B \leq q_k \leq B \qquad\qquad\qquad\quad (k = 1, \ldots, d), \\ & r_{\text{LP}} \geq \tau_{\text{LP}}. \end{aligned} \tag{15}$$

We use $B = 100$ in the experiments and set $\tau_{\text{LP}} = 10^{-8}$. If the LP is infeasible, $\mathcal{R}_t(E)$ is empty, then the target is irretrievable by any query. If it is feasible, $\mathcal{R}_t(E)$ is nonempty, and a ball of radius r_{LP} fits in the LP region. r_{LP} is thus a margin which measures the size of the region, but does not correspond to the cone volume specifically.

D.2 Hull projection by Frank–Wolfe

To project a point $p \in \mathbb{R}^d$ onto $P = \text{conv}\{g_1, \ldots, g_M\}$, we write $x = \sum_j w_j g_j$ with $w_j \geq 0$ and $\sum_j w_j = 1$, and minimise $\|x - p\|_2^2$ by away-step Frank–Wolfe (Frank & Wolfe, 1956; Lacoste-Julien & Jaggi, 2015). All weight starts on $j_0 = \arg\max_j \langle g_j, p\rangle$. Each iteration multiplies the

$M \times d$ matrix with rows g_j by the residual $x - p$, giving $\langle g_j, x - p\rangle$ for every j, and picks

$$j^+ = \arg\min_j \langle g_j, x - p\rangle, \qquad j^- = \arg\max_{j:\, w_j > 0} \langle g_j, x - p\rangle. \tag{16}$$

It then moves x along $h = g_{j^+} - x$ or $h = x - g_{j^-}$, whichever has the smaller $\langle x - p, h\rangle$. The step $x \leftarrow x + \gamma h$ uses the exact line search $\gamma = \langle p - x, h\rangle / \|h\|_2^2$, clipped to $[0, 1]$ towards g_{j^+} and to $[0, w_{j^-}/(1 - w_{j^-})]$ away from g_{j^-} so that no weight turns negative. Correspondingly, all weights are scaled by $1 - \gamma$ and w_{j^+} increased by γ, or scaled by $1 + \gamma$ and w_{j^-} decreased by γ. We stop once the duality gap $2\langle x - p, x - g_{j^+}\rangle$, an upper bound on the excess of $\|x - p\|_2^2$ over its minimum, is below 10^{-12}, once $\gamma = 0$, or at an iteration cap.

D.3 Target-only optimisation

For the target-only experiments in Section 3.3, the initial Adam learning rate is 0.02, with cosine decay to zero. Squared and unsquared hull distance use an adaptive Frank–Wolfe projection, recomputed every 25 steps with up to $1{,}200$ iterations.

We choose the optimised image with the smallest recorded Frank–Wolfe residual, stopping early if an in-loop LP certifies the target hull containment. This early stopping check runs every 100 steps once the best residual falls below 2% of its initial value, with a 30-second timeout budget. Final classification uses the re-encoded input and a 900-second LP budget with at most $20{,}000$ iterations against all other keys.

D.4 Spectral knee selection

To select the spectral knee for coordinate rescaling in Equation (7), let $a_k = (\log \sigma_k - \log \sigma_d)/(\log \sigma_1 - \log \sigma_d)$ and $b_k = (k - 1)/(d - 1)$ for $k = 1, \ldots, d$. The spectral-knee rule chooses $j_* = \arg\max_k \{a_k - (1 - b_k)\}$ and sets $k_* = d - j_*$.

D.5 Decoy initialisation

For a given $\widetilde{z}$, we select C images from the retrievable item set as decoy initialisations with the solver of Appendix D.2, taking $p = \widetilde{z}$ and the g_j to be the rescaled embeddings of these images. The first image is j_0, and each of the next $C - 1$ steps adds the image j^+ of Equation (16) and moves x towards its embedding. We take no away steps and exclude already selected images from j^+. The weights are recomputed after selection.

Let $F_i = T(f(\xi_i))$ denote the embeddings of the selected image initialisations. We solve the regularised simplex problem

$$\min_{w \in \Delta^{C-1}} \left\| \widetilde{z} - \sum_{i=1}^{C} w_i F_i \right\|_2^2 + \lambda_{\text{ridge}} \nu \sum_{i=1}^{C} w_i^2, \tag{17}$$

where $\nu = C^{-1} \sum_{i=1}^{C} \|F_i - \widetilde{z}\|_2^2$. The ridge penalty with $\lambda_{\text{ridge}} = 10$ avoids over-concentration of the weights.

D.6 Decoy optimisation

With the weights fixed, the Jacobian of the weighted mean in Equation (8) with respect to all decoy pixels is $[w_1 J_1, \ldots, w_C J_C]$, where $J_i \in \mathbb{R}^{d \times D}$ is the Jacobian of $T \circ f$ at the quantised decoy input. Its Gram matrix G defines the metric

$$G = \sum_{i=1}^{C} w_i^2 J_i J_i^\top, \qquad M = \Lambda \left(G + \gamma_{\text{G}} \Lambda I_d\right)^{-1}, \tag{18}$$

where $\Lambda = \lambda_{\max}(G)$ and $\gamma_{\text{G}} > 0$ is a damping factor. A gradient step on the M-weighted loss shrinks the residual at about the same rate along every eigenvector of G whose eigenvalue is well above $\gamma_{\text{G}} \Lambda$. Under the unweighted loss, this rate is proportional to the eigenvalue instead, so directions in which the mean responds weakly to the pixels are barely corrected. M is recomputed at the current iterate at fixed intervals and held constant.

Probe estimator. Forming G exactly requires d backward passes per decoy. Instead, at each recomputation we draw fresh pixel probes $g_{i\ell} \sim \mathcal{N}(0, I_D)$ and compute M from

$$\widehat{G} = \frac{1}{n_{\text{probe}}} \sum_{i=1}^{C} w_i^2 \sum_{\ell=1}^{n_{\text{probe}}} (J_i g_{i\ell})(J_i g_{i\ell})^\top, \tag{19}$$

which is unbiased for G since $\mathbb{E}[g_{i\ell} g_{i\ell}^\top] = I_D$, and costs n_{probe} Jacobian–vector products per decoy.

Optimisation settings. We use $\gamma_{\text{G}} = 10^{-4}$ and $n_{\text{probe}} = 32$. For other values of C, we keep the total $Cn_{\text{probe}} = 3{,}072$, which exceeds d for every encoder. The perturbations start from $\delta_i = 0$ and are optimised by projected Adam with initial learning rate 0.02, $\beta_1 = \beta_2 = 0.9$ and cosine decay to zero. M is recomputed twelve times per run every 125 steps.

D.7 Normalised-scoring experiments

Correction of shared directional error. To correct collective tangent imbalance, we centre the realised keys' unit tangent directions, reproject onto $\bar{e}_t^{\perp}$, and renormalise. We iterate until the mean direction has norm below 10^{-9} or 50 iterations have elapsed, then reconstruct keys at their original target–decoy angles. This changes embeddings directly; the corrected keys have not been realised as images.

Only the MobileCLIP2 $C = 64$ entry in Table 3 uses this correction. All decoy rows in Table 5 use the same correction.

Decoy count and output dimension ablation. Figure 4b) reduces the output dimensions of MetaCLIP 2 to $d \in \{1024, 768, 512, 256, 128\}$ by PCA at $\alpha = 5°$, and correspondingly reduce the decoy count C to hold the cone volume roughly fixed: $C = 8, 7, 6, 6, 5$ at 2^{-10}, $16, 14, 12, 11, 10$ at 2^{-21} and $32, 28, 25, 23, 20$ at 2^{-43}. Similarly for panels (c,d), to maintain a fixed cone volume, increases in C are matched with increases in α: $(C, \alpha) = (16, 2.47°)$, $(32, 8.17°)$, $(64, 15°)$, $(128, 20°)$, and $(256, 24.54°)$.

D.8 Query search

Text query search. ARCA text query search minimises

$$\mathcal{L}_{\text{search}}(q) = \log \sum_{j \neq t} \exp\big(\ell_j(q)\big) - \ell_t(q),$$

where q is the encoded query and $\ell_j(q)$ its scaled cosine logit against key j in the augmented key set. At each position, a linear approximation using gradients averaged over four random token and shortlists 32 candidates for exact scoring. We return the first rank-one sequence or the smallest-loss sequence at budget exhaustion. Returned sequences are decoded and re-tokenised before evaluation.

Image query search. For Table 5, the search optimises a query image to retrieve the target under the image-to-image settings. Query pixels are initialised uniformly in $[0.15, 0.85]$ and kept in the valid range $[0, 1]$ by a sigmoid parameterisation, so the search can reach any valid image. Adam with learning rate 0.03 minimises the text-search objective $\mathcal{L}_{\text{search}}$ for up to $10{,}000$ steps, stopping at the first step where the target ranks first.

D.9 Runtime

Each run uses a single NVIDIA GH200 GPU. We report the median wall-clock time per target. For decoy insertion in Table 1, a $1{,}500$-step run takes 8, 25, 12, 31, 65 and 97 minutes on ViT-B/32, RN101, SigLIP 2 B/16, MobileCLIP2, MetaCLIP 2 and SigLIP 2 SO400M. Target-only perturbation takes 3–26 minutes per objective. In Section 4.2, realising $C = 64$ decoys takes $3.5 - 28$ minutes on these six encoders, 57 minutes on Qwen3-VL-Embedding-2B, 18 minutes on DINOv2 and 10 minutes on DINOv3.

E ADDITIONAL ABLATION STUDIES

E.1 HULL CONTAINMENT IN NATURAL IMAGE DATASETS

We test every image of four image datasets, Flickr30K (Young et al., 2014), MS-COCO (Lin et al., 2014), ImageNet-1k (Deng et al., 2009) and Open Images V7 (Kuznetsova et al., 2020), under ViT-B/32, with the whole dataset taken as the retrievable set. As shown in Table 6, we find no image that lies inside the hull of the others, meaning there is no naturally occurring top-1 irretrievable image in these datasets.

Table 6: Images whose ViT-B/32 embedding lies inside the convex hull of the other images.

Dataset	Images	Inside hull
Flickr30K	31,014	0
MS-COCO 2017, train + val	123,287	0
ImageNet-1k, train	1,281,167	0
Open Images V7, box subset	1,910,098	0

E.2 RANDOM PERTURBATION OF THE TARGET

We test whether adding random noise to the target image alone can make it irretrievable on ViT-B/32. Every pixel of the target is shifted by at most $\pm\varepsilon$ at random, and all other items are left unchanged. For each ε, we sample 10^6 copies per target and check for hull containment. As shown in Table 7, no embedding lies inside the hull at any ε.

Table 7: The number of hull-contained images with one million copies of 12 randomly noised target images on ViT-B/32.

ε	Inside hull
$2/255$	0
$4/255$	0
$8/255$	0
$16/255$	0
$32/255$	0

E.3 HULL CONTAINMENT AT THE MINIMUM ITEM COUNT

In this appendix we test the algorithm's performance near the minimum number of items required to form a nonzero-volume hull. In d dimensions, this means there must be at least $d + 1$ vertices. Under decoy insertion, the hull that must contain the target is formed by the other $N - 1$ retrievable items and the C decoys, so the construction cannot succeed unless $N-1+C \geq d+1$. We rerun the ViT-B/32 setting of Figure 3 c) at or near the floor $N = 418$, with the items randomly sampled from Flickr30K. Figure 5 shows that 4 of the 24 targets are hidden even at the minimum required number of items for forming a nonzero-volume hull, which again demonstrates the impressive performance of the algorithm under this extreme case.

E.4 LARGER RETRIEVABLE SETS

We rerun the ViT-B/32 experiments of Tables 1 to 3 with three larger image datasets as the retrievable set: MS-COCO 2017 train and validation (Lin et al., 2014), ImageNet-1k train (Deng et al., 2009) and Open Images V7 (Kuznetsova et al., 2020). For each dataset, 48 targets are sampled uniformly at random, and all other settings follow Sections 3.3 and 4.2. ImageNet-1k and Open Images V7 have no human-written captions, so caption rank is reported on MS-COCO only.

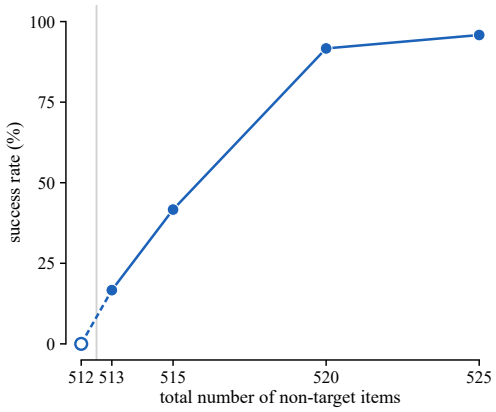

Figure 5: Success rate of decoy insertion on ViT-B/32 against the number of non-target items $N - 1 + C$. The grey vertical line marks the minimum item threshold.

As shown in Table 8, decoy insertion again achieves near-perfect hidden rates on the three larger datasets. Target-only perturbation remains weak on every dataset. On MS-COCO, the median caption rank starts higher than on Flickr30K, which has a quarter as many images, but the decoys raise it by the same median amount on both datasets: 4 places at $C = 8$ and 29.5 at $C = 64$. Under ARCA text search, $C = 64$ hides every target on every dataset.

Table 8: Results of Tables 1 to 3 on ViT-B/32 on different datasets.

		Flickr30K $N = 31{,}014$	MS-COCO $N = 123{,}287$	ImageNet-1k $N = 1{,}281{,}167$	Open Images V7 $N = 1{,}910{,}098$
Empty cone	Squared hull distance	0.0%	0.0%	0.0%	0.0%
	Unsquared hull distance	0.0%	2.1%	2.1%	0.0%
	Hull centroid	18.8%	8.3%	10.4%	14.6%
	Decoy insertion	**100.0%**	**100.0%**	**97.9%**	**100.0%**
Caption rank	$C = 0$	3	48	–	–
	$C = 8$	7	53	–	–
	$C = 64$	**33**	**80.5**	–	–
Text search	$C = 0$	12.5%	33.3%	33.3%	37.5%
	$C = 8$	66.7%	52.1%	60.4%	77.1%
	$C = 64$	**100.0%**	**100.0%**	**100.0%**	**100.0%**

E.5 Construction angle and output dimension

We rerun the 2^{-43} line of Figure 4b) with $C = 32$, and fix the cone volume by raising α instead: $5°$, $5.89°$, $6.94°$, $7.80°$ and $9.34°$ at $d = 1024$, 768, 512, 256 and 128. As shown in Figure 6, below $d = 512$, lowering C keeps every target hidden, whereas raising α reduces the hidden rate, again indicating that α is a stronger factor than C. Above $d = 512$, raising dimension reduces the hidden rate for both increasing C or lowering α, indicating the output dimension is a stronger factor.

E.6 Image-to-text and text-to-text retrieval

We apply the decoy construction of Section 4.1 to image-to-text and text-to-text retrieval. The retrievable set is the 155,070 Flickr30K captions under ViT-B/32, and each target is the first caption of one of the 48 images of Section 4.2. We use $C = 16$ decoys at $\alpha = 5°$ and realise each as 16 tokens optimised with ARCA for 1,200 coordinate updates, then decoded and re-tokenised. The queries are either the image the target caption describes or the four other captions of the same image.

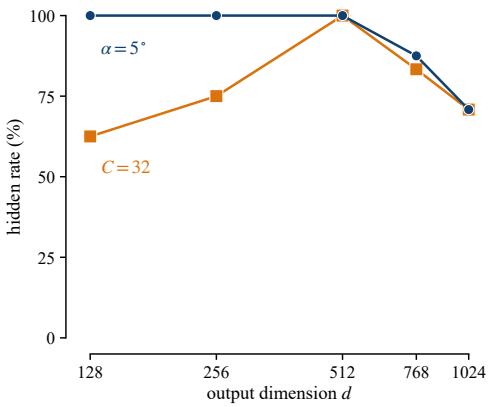

Figure 6: Hidden rate on MetaCLIP 2 after PCA reduction to d output dimensions, with the cone volume held at 2^{-43} either by lowering C at $\alpha = 5°$ as in Figure 4b), or by raising α at $C = 32$.

As shown in Table 9, the no-op retrieval rate is generally lower than other sections as there are five times more retrievable items in the dataset. The inserted decoys lower the retrieval rate for both retrieval types, but only modestly. We attribute this mostly to the limited capability of the underlying discrete token optimisation algorithm. The realised decoys lie a median $26.4°$ from the target instead of the designed $5°$.

Table 9: Retrieval rate of the target caption under text decoy insertion.

	Image-to-text		Text-to-text	
Inserted decoys	Top-5 Ranking	Top-10 Ranking	Top-5 Ranking	Top-10 Ranking
No-op	31.2%	41.7%	14.6%	19.8%
Text decoys	25.0%	33.3%	8.9%	13.5%