CONCEPT-BLOBS: THE GEOMETRY OF CONCEPTS AND ATTRIBUTE BINDING IN FOUNDATION MODELS

Anonymous authors
Paper under double-blind review

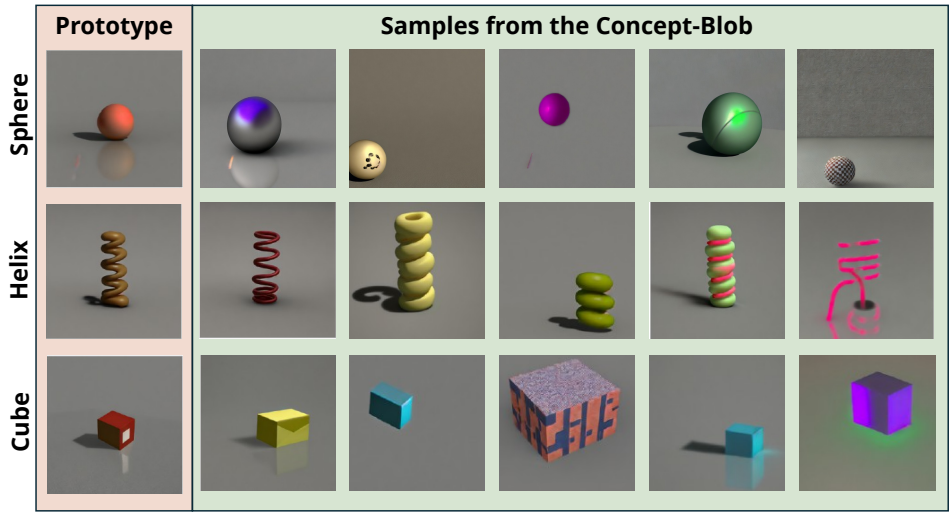

Figure 1: **Concept-Blobs.** Left: Concept prototype μ_c. Right: Random samples drawn from the Concept-Blob. While the prototype represents a canonical object, typically centralized and with fixed attributes; samples from across the blob exhibit wide variations in lighting, position, size, color, and orientation, capturing the full richness of concept variability. The examples above are generated from the CLIP embedding space using unCLIP (Ramesh et al., 2022). For DINO, SigLIP2, and AIMv2, similar intra-blob attribute representation is validated via linear probing.

ABSTRACT

How are concepts geometrically organized within foundation-model representation space? How can an attribute, such as color, be bound to a specific object? While existing literature often models concepts as one-dimensional linear directions or as a decomposition into sparse features, we propose a paradigm shift by modeling the geometry as *Concept-Blobs*. A blob of a concept c has a compact effective support, parametrized by a mean vector and a covariance matrix (μ_c, Σ_c), inducing a Mahalanobis metric within the blob. Rather than collapsing a concept into a single vector or an unconstrained manifold, Concept-Blobs explicitly capture intra-concept variance across attributes such as position, lighting, pose, and color. We show that this approach models multi-concept scenes significantly better. We model concept composition as a scaled Minkowski sum of individual blobs. Across various models in both visual and textual domains (CLIP, DINOv3, AIMv2, and SigLIP2 for images; CLIP, SigLIP2, Qwen3-Embedding, and Sentence-T5 for text), we reveal two key observations in a fully controlled multi-concept setting based on CLEVR: (1) the residual information left after composing individual concepts, which we call representation *dark matter*, can be largely explained by inter-concept relationships through a dedicated relational "and" blob; and (2) attribute binding emerges naturally from the geometry, associating attributes with their corresponding objects without requiring explicit binding constraints. These results suggest that foundation-model representations encode

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.

concepts not as isolated directions, but as structured probabilistic regions whose geometry supports both relational composition and attribute binding.

1 INTRODUCTION

Current geometric understanding. Foundation models, such as CLIP (Radford et al., 2021) and DINO (Siméoni et al., 2025), increasingly serve as general-purpose representations for modern vision and vision-language systems. Understanding how concepts are geometrically embedded within these latent spaces is therefore of critical importance, as their structure directly governs how information is represented, related, and composed, with implications that extend across a wide range of downstream tasks (Ramesh et al., 2022; Patashnik et al., 2021; Belrose et al., 2023; Arditi et al., 2024; Trager et al., 2023; Berasi et al., 2025). Recent efforts have made significant advancements in characterizing the global geometry of these spaces, uncovering structural phenomena like the modality gap and narrow cone effect (Liang et al., 2022; Schrodi et al., 2025), alongside broader geometric (Levi & Gilboa, 2025; Yu et al., 2026), probabilistic (Betser et al., 2025; Tasker et al., 2026), and theoretical (Betser et al., 2026; Cai et al., 2026) frameworks. These global perspectives offer little insight into how individual concepts are structured locally. While fine-grained concept analysis has gained traction, most prominently through sparse autoencoders (SAEs) (Huben et al., 2024; Elhage et al., 2022; Engels et al., 2025) and dictionary learning (Bhalla et al., 2024; Gordon et al., 2026), these approaches remain primarily decomposition-driven, prioritizing metrics such as reconstruction or sparsity over characterizing the intrinsic geometry of the local concept space itself. Similarly, recent investigations into attribute binding and compositional behavior (Koishigarina et al., 2026; Uselis et al., 2026b) highlight the limitations of purely additive composition of concepts, yet they continue to model concepts as rigid vectors. Thus, they leave fundamental questions unanswered, for example: *how do individual concepts span the latent space, and how are combinations of concepts geometrically organized?*

From single directions to structural volumes. A common attractive underlying assumption is that concepts are modeled as single representative directions in space. If both objects and attributes are single linear directions what happens when there is more than one object? For example, if we have linear arithmetic of a scene embedding by $(scene) = \{Apple, Ball, Green\} = Apple + Ball + Green$, which object is green, the ball or the apple? Binding in this case is impossible. However, we already know that foundation models preserve binding information *within each modality*: object-level information is recoverable by a probe inside the image or the text encoder alone, even where cross-modal retrieval fails (Koishigarina et al., 2026; Uselis et al., 2026b). Such a basic model therefore appears too simplistic. We argue that a single vector inherently fails to capture the internal variability of a concept: when a human considers "a dog" for example, the mental representation encompasses rich variations across breeds, poses, and viewpoints that a single direction will fail to encapsulate. A step forward in addressing this limitation is the recent work by Fel et al. (2026), which models concepts as non-linear manifolds. However, because these manifolds are learned implicitly through training, they lack explicit statistical characterization, thus, difficult to interpret.

Introducing Concept-Blobs. In this work, we propose a paradigm shift: concepts in foundation models are neither isolated points, rigid linear directions, nor arbitrary black-box manifolds, but structurally coherent *Concept-Blobs*. We posit that a concept c intrinsically occupies a probabilistic volume within the latent space, modeled explicitly by both its mean vector μ_c and its covariance matrix Σ_c, which captures its natural spread, diversity, and internal characteristics. Intuitively, while the mean μ_c anchors the canonical concept, e.g., generic "dog ", the displacement $U_c\alpha_c$ navigates within the blob's geometric bounds to capture instance-specific nuances; such as breed, pose, or lighting, without straying into implausible representations. See illustration in Fig. 1 showing the mean and several random examples of samples from several concepts. This probabilistic formulation naturally extends from single concepts to complex representations. For a scene embedding containing a set of co-occurring concepts, we model the composed embedding as a *scaled Minkowski sum* of their respective blobs. Under this view, composition is not a fixed, rigid vector addition. Instead, it is an adaptive selection process: each participating concept contributes an instance-specific vector, selected from within its valid volume.

Resolving "dark matter". We evaluate our scaled Minkowski-sum model on the CLEVR dataset (Johnson et al., 2017), which provides ground-truth specifications for every object and attribute in

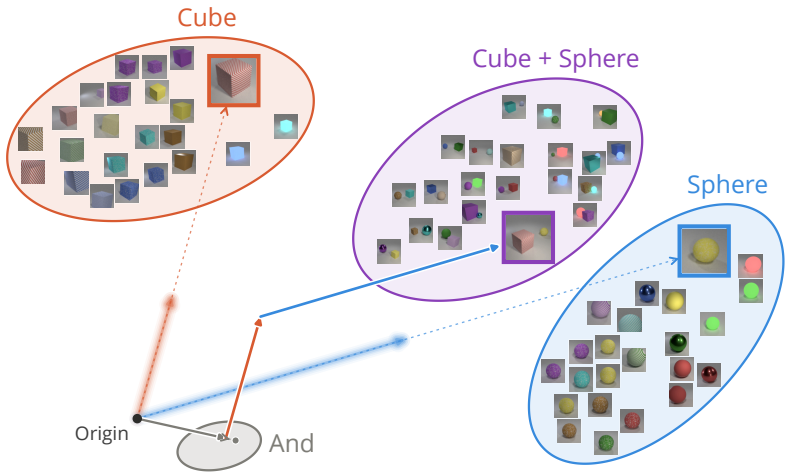

Figure 2: **Scaled Minkowski sum.** a composition of several objects, each with its own attributes, is a scaled sum of its constituent objects.

a scene. Surprisingly, even when every constituent object and physical attribute is isolated and accounted for, composing an embedding strictly from individual object blobs leaves a non-negligible residual gap to the original scene embedding, which we call representation *dark matter*, borrowing the term from sparse autoencoders (Engels et al., 2025). We hypothesize that this missing information reflects inter-object relationships (e.g., "in front of", "beside" and "to the left of"). In our framework, this relational richness is captured by introducing a dedicated relational blob, $\mathcal{B}_{\text{and}}$, whose covariance explicitly models multi-object interactions. Incorporating $\mathcal{B}_{\text{and}}$ yields near-perfect reconstruction fidelity, suggesting that inter-concept relationships may account for this residual.

Emergent attribute binding through geometry. A second striking observation is that optimizing over Concept-Blobs without explicit binding constraints or regularizers automatically binds attributes to their corresponding objects with high fidelity (Table 1). This demonstrates that attribute binding is an emergent property of the latent geometry itself. While the Linear Representation Hypothesis (Park et al., 2024) assumes single global directions, high-dimensional space affords a vast directional aperture. Unlike SAEs and dictionary learning, which select fixed single directional features without an intrinsic binding mechanism, selecting a point within a Concept-Blob inherently performs attribute binding, as every vector inside the volume represents a concrete, attribute-contextualized instance. This is still an open research issue, however as we currently understand it, the same attribute is encoded differently when describing different object. Going back to our *apple and ball* example, there are two "flavors" of green, for a green ball we can schematically view this as $(scene) = \{Apple, Ball, Green_{ball}\} = Apple_{\mu} + (Ball_{\mu} + Green_{ball})$, where μ denotes the mean of the concept and the ball has a richer description from its blob. Note that generally $Green_{ball}$ and $Green_{apple}$ are different, hence binding of attributes to specific objects is possible. A few examples from our controlled experiments, where the same attributes are represented differently for different objects, are seen in Fig. 9 at the appendix.

Extensive ablation study. We evaluate the Concept-Blob framework across 9 frozen encoders spanning pre-training paradigms: contrastive vision-language models (CLIP, SigLIP2), purely visual self-supervision (DINOv3 B/16 and L/16), and generative pre-training (AIMv2), so that nothing we report rests on a single training objective. The framework also scales beyond pairs: the same fit recovers per-object attributes on three-object scenes, and on real images assembled from ImageNet crops, where per-object ground truth is known by construction. Furthermore, we demonstrate that this geometric behavior is not exclusive to the visual modality. We show that across four text encoders (CLIP, SigLIP2, Qwen3-Embedding and Sentence-T5) these representations encapsulate probabilistic variations, functioning as text-space concept-blobs that capture semantic diversity.

Our main contributions are:

- **Concept-Blob Formulation:** We introduce a probabilistic geometric model for concepts, defining them as compact support regions, parameterized by their mean and covariance.
- **Scaled Minkowski-Sum Composition:** We model multi-concept composition as a scaled Minkowski sum of concept blobs
- **Optimization Scheme:** We propose an optimization mechanism to decompose an embedding of multiple objects into its components, yielding SOTA attribute binding.
- **Relational Blob and Dark Matter:** We demonstrate that introducing a dedicated relational blob, significantly reduces the residual reconstruction gap, suggesting that *dark matter* can be explained by inter-object relationships.
- **Emergent Attribute Binding:** We show that foundation models naturally solve the feature-binding problem by partitioning global attribute directions into object-specific subdirections without explicit external constraints.
- **Extensive Validation of Model Generality:** We establish the generality of Concept-Blobs across nine encoders spanning self-supervised vision (DINOv3), generative pre-training (AIMv2), vision-language spaces (CLIP, SigLIP2) and dedicated text embedders (Qwen3-Embedding, Sentence-T5). In addition, we extend the model to three-object scenes and to real images.

2 Related Work

Concepts as directions. Most prior work represents a concept as a single vector and a combination of concepts as a sum of their vectors. This began with word-vector arithmetic (Mikolov et al., 2013), was formalized as the linear representation hypothesis (Park et al., 2024; Jiang et al., 2024), and reached vision-language models as "ideal words" (LDE): Trager et al. (2023) approximate CLIP text embeddings as an anchor plus one vector per factor, "a [color] [object]" $\approx u_0 + u_{\text{color}} + u_{\text{object}}$. Berasi et al. (2025) carry the same factorization to image embeddings on the unit sphere. Further work ties this additive structure to compositional generalization (Abbasi et al., 2025; Uselis et al., 2025; 2026a). Our model departs from the works above on two main issues: a concept occupies a region whose interior carries its attribute variation, and attribute displacement is object-dependent.

Binding. The binding problem asks how a representation records which attribute belongs to which object (Treisman & Gelade, 1980; Greff et al., 2020). In cross-modal retrieval CLIP appears to fail the binding problem, behaving as a bag of words (Yuksekgonul et al., 2023; Tang et al., 2023; Lewis et al., 2024; Campbell et al., 2024; Gurung et al., 2025; Kang et al., 2025; Zhang et al., 2025). The information is nevertheless present (Li et al., 2025): Koishigarina et al. (2026) recover binding with a probe inside each encoder separately, and Uselis et al. (2026b) provide justifications for such an approach.

Manifolds, sparse features, and residuals. Sparse autoencoders and dictionary learning (Elhage et al., 2022; Huben et al., 2024; Bhalla et al., 2024; Gordon et al., 2026) write an embedding as a sparse combination of fixed directions, so a concept is again a single atom. Fel et al. (2026) instead model activations as a Minkowski sum of concept *manifolds*, learned implicitly through a block-sparse featurizer. We adopt the Minkowski view but make each region explicit: a mean, a covariance, and a support calibrated from labeled samples, fitted per composite by a convex program and tested by decomposing held-out composites. Finally, the part of an embedding that such models fail to reconstruct has been called representation *dark matter* (Engels et al., 2025); we use the term for what additive composition of the true constituents cannot explain, and trace part of it to the "and" blob itself.

3 The Concept Blob Model

3.1 Definition

We assume to have a d-dimensional embedding space $X \subset \mathbb{R}^d$, where an embedding instance $x \in X$ can be any world concept or combination of concepts and their attributes. We do not constrain ourselves to a specific modality, and assume that some pretrained foundation model F provides a d-dimensional representation vector $x = F(g)$, for any instance g of a certain modality.

Table 1: **Attribute binding against composition baselines.** Conducted on CLIP ViT-L/14, where *both* refers to the fraction of composites with both colors and materials bound to the correct objects. Concept-Blobs alone already surpass all baselines; adding the "and" blob widens the gap.

		Image			Text		
Method	Composition	color	material	both	color	material	both
LDE (Trager et al., 2023)	$\hat{e} = c_0 + \sum_i u_{a_i} + \sum_i u_{b_i}$	0.828	0.855	0.555	0.641	0.809	0.489
GDE (Berasi et al., 2025)	$\hat{e} = \mathrm{Exp}_\mu(v_a + v_b)$	0.782	0.850	0.490	0.638	0.789	0.458
Single-obj (Uselis et al., 2026b)	$\hat{e} = \hat{u}_a + \hat{u}_b$	0.823	0.890	0.605	0.639	0.694	0.407
Concept-Blobs w/o "and"	$\hat{e} = w_a x_a + w_b x_b,\ x_c \in \mathcal{B}_c$	0.840	0.940	0.650	0.775	0.980	0.625
Concept-Blobs (ours)	$\hat{e} = \sum_c w_c x_c,\ x_c \in \mathcal{B}_c$	**0.875**	**0.960**	**0.710**	**0.840**	**0.998**	**0.715**

The derivation is modality-free, and thus holds for other modalities such as audio or 3D, although this paper examines text and images.

A concept is some semantic entity (like a ball, a chair, a dog) that can have many variations and attributes, and can reside and interact with other concepts. We assume a concept c as a statistical object. It thus has a mean $\mu_c \in \mathbb{R}^d$ and a covariance matrix $\Sigma_c \in \mathbb{R}^{d\times d}$ which reflect its probability density function in X over all possible variations, attributes and interactions. Let $U_c \in \mathbb{R}^{d\times k}$ denote the top-k principal directions (eigenvectors) of Σ_c and $\Lambda_c \in \mathbb{R}^{k\times k}$ the diagonal matrix of the corresponding eigenvalues, i.e. the variance values along these directions (for $k = d$, $\Sigma_c = U_c\Lambda_c U_c^\top$). Then, for a given concept c, let $\mathcal{B}_c$ denote its *concept-blob*, defined as the concept's essential probabilistic support:

$$\mathcal{B}_c := \left\{ x_c \in X \ :\ x_c = \mu_c + U_c\alpha_c, \quad \alpha_c \in \mathbb{R}^k, \quad \|\Lambda_c^{-1/2}\alpha_c\| \le \tau_c \right\}, \tag{1}$$

where α_c controls the extent of displacement at each principal direction, and τ_c controls the support to be compact, based on the Mahalanobis distance from μ_c. The ambiguous question of *what is a concept* is replaced by an explicit membership test: does a sample fall inside the blob region?

3.2 Multiple Concepts as a Scaled Minkowski Sum

Real-world scenes typically contain multiple concepts. Thus, knowing the statistics of single-object blobs in advance is insufficient. We now ask how multiple concepts span the space when they appear together. Inspired by Fel et al. (2026) and the broadly linear behavior of embedding spaces (Mikolov et al., 2013; Park et al., 2024), a natural first step for modeling multi-concept scenes would be the Minkowski sum of their constituent blobs. However, a standard Minkowski sum assumes unscaled combinations, whereas embedding norms in self-supervised learning (SSL) are known to correlate with frequency or uncertainty (Draganov et al., 2025; Su et al., 2026). For instance, while an airplane and a fish are individually common concepts, their composition forms an extremely rare scene; the representation space should reflect this shift in rarity through its magnitude.

This motivates our formulation to propose the *scaled Minkowski sum* (illustrated in Fig. 2). Recall that the Minkowski sum of two sets is $A \oplus B := \{a + b \ :\ a \in A, b \in B\}$, and write $wA := \{wa \ :\ a \in A\}$ for the dilation of a set by a scalar $w > 0$. Each concept $c \in \mathcal{S}$ present in the scene contributes a point from its blob, scaled by a weight w_c. For a fixed weight vector $w = (w_c)_{c\in\mathcal{S}}$ this defines the *multi-concept blob*

$$\mathcal{B}_\mathcal{S}(w) \ := \ \bigoplus_{c\in\mathcal{S}} w_c\,\mathcal{B}_c \ = \ \left\{ \textstyle\sum_{c\in\mathcal{S}} w_c x_c \ :\ x_c \in \mathcal{B}_c \ \ \forall c \in \mathcal{S} \right\}. \tag{2}$$

Our model assumes that the embedding e of a multi concept scene admits $e \in \mathcal{B}_\mathcal{S}(w)$, yielding:

$$e \ = \ \sum_{c\in\mathcal{S}} w_c\, x_c, \qquad x_c \ \in \ \mathcal{B}_c, \qquad \forall c \in \mathcal{S}, \tag{3}$$

where $\mathcal{B}_c$ is the concept-blob of concept c , Equation (1).

3.3 THE RELATIONAL "AND" BLOB

Our model assumes that in multi-concepts scenes an additional relational concept exists that captures the joint co-occurrence of both concepts in a single scene, acting as a geometric "image-equivalent" to the word "and" in text. Because this relational term can encode high-level interactions like "beside" or "on top of", we treat it the same as any other concept. Thus we introduce a new special concept blob $\mathcal{B}_{\text{and}}$, with its mean μ_{and}, covariance Σ_{and}, and radius τ_{and}. For instance, for a dual object scene, Equation (3) becomes $e = w_1 x_1 + w_2 x_2 + w_{\text{and}}\, x_{\text{and}}$, with $x_c \in \mathcal{B}_c$ for $c \in \{1, 2, \text{and}\}$.

4 DECOMPOSITION BASED ON THE BLOB MODEL

We now use the blob model developed above to perform concept decomposition of a multi-object scene. We first need to estimate the parameters of each concept blob.

4.1 PARAMETER ESTIMATION

For estimating the statistics $(\mu_c, \Sigma_c, \tau_c)$ of Section 3, we use n_c embeddings $\{z_1, \ldots, z_{n_c}\}$ of instances that contain concept c alone, varied over its attributes (see details in Section E). We compute the empirical mean and covariance,

$$\hat{\mu}_c = \frac{1}{n_c}\sum_{i=1}^{n_c} z_i, \qquad \hat{\Sigma}_c = \frac{1}{n_c - 1}\sum_{i=1}^{n_c}(z_i - \hat{\mu}_c)(z_i - \hat{\mu}_c)^\top = \hat{U}_c \hat{\Lambda}_c \hat{U}_c^\top, \tag{4}$$

with eigenvalues in decreasing order. We take the top-k eigenvectors and their eigenvalues, $\hat{U}_c \in \mathbb{R}^{d\times k}$ and $\hat{\Lambda}_c \in \mathbb{R}^{k\times k}$. Each instance has Mahalanobis distance from the mean of $\rho_i = \|\hat{\Lambda}_c^{-1/2}\hat{U}_c^\top(z_i - \hat{\mu}_c)\|$. For stability reasons we set $\hat{\tau}_c$ to the 95th percentile of $\{\rho_i\}_{i=1}^{n_c}$, the empirical radius that contains 95% of the concept's samples. We select the distribution of samples primarily by the object, as we observe a clear hierarchy, as shown in the appendix, see Fig. 10.

Samples of the "and" blob $\mathcal{B}_{\text{and}}$ cannot be observed directly. We estimate them from composites e whose single-concept embeddings e_1, e_2 are known, taking the residual of the best unconstrained weighted sum,

$$r = e - (w_1^\star e_1 + w_2^\star e_2), \qquad (w_1^\star, w_2^\star) = \arg\min_{w_1, w_2} \|e - w_1 e_1 - w_2 e_2\|_2^2. \tag{5}$$

Based on the residuals r of many such concept pairs we compute the estimations $\hat{\mu}_{\text{and}}$, $\hat{U}_{\text{and}}$, $\hat{\Lambda}_{\text{and}}$ and $\hat{\tau}_{\text{and}}$ exactly as above (see more details in Section E). We examined the meaning of $e_1 + e_{\text{and}}$, that is, appending "and" to a single object. This appears to signify multiplicity, usually adding a tiny new object adjacent to the original object, as demonstrated by the synthesis shown in Fig. 7.

4.2 DECOMPOSITION BY CONVEX OPTIMIZATION

We treat decomposition as an inverse problem, attempting to invert a given multi-concept embedding e into its components, based on our blob model, Equations (1) and (3). Fitting the composition model to an embedding e requires jointly optimizing the region points x_c and their weights w_c over all present concepts and the relational blob, $\mathcal{S}' = \mathcal{S} \cup \{\text{and}\}$. Parameterizing this fit in (w_c, α_c) yields a non-convex bilinear objective due to the product $w_c \hat{U}_c \alpha_c$, offering no global optimality guarantees. We resolve this non-convexity via the change of variables $\beta_c := w_c \alpha_c$. This substitution linearizes the reconstruction term into $w_c\hat{\mu}_c + \hat{U}_c\beta_c$ and transforms each support region into an exactly equivalent second-order cone constraint, jointly convex in (β_c, w_c) for $w_c \geq 0$. Model fitting thus reduces to a single Second-Order Cone Program (SOCP):

$$\min_{\{w_c, \beta_c\}} \left\| e - \sum_{c\in\mathcal{S}'} \left(w_c\hat{\mu}_c + \hat{U}_c\beta_c\right)\right\|_2^2 \quad \text{s.t.} \quad \|\hat{\Lambda}_c^{-1/2}\beta_c\| \leq \hat{\tau}_c w_c, \;\; w_{\min} \leq w_c \leq w_{\max} \;\; \forall c \in \mathcal{S}', \tag{6}$$

which is solved globally in a single step using standard conic solvers without alternation. Since the exact weights of equation 2 are not known in advance, we optimize them over a range $[w_{\min}, w_{\max}]$. The full derivation of this convex reformulation, along with the choice of these bounds, the weights they produce and solver details, is given in Section C.

Table 2: **Does anonymization remove identity completely?** Subtracting the concept mean removes overall shape identity ($\approx 7\%$). However, when holding values constant (evaluating per material or color value), shape identity becomes fully recoverable ($\approx 99\%$), confirming that attribute displacements remain concept-dependent. (See t-SNE visualization in Fig. 9)

			CLIP ViT-L/14		DINOv3 ViT-B/16		DINOv3 ViT-L/16	
attribute	**C**	**chance**	raw	anon	raw	anon	raw	anon
color	7	0.143	1.000	1.000	0.998	0.999	1.000	1.000
material	5	0.200	0.995	0.995	1.000	0.999	1.000	1.000
shape	11	0.091	0.999	**0.075**	1.000	**0.073**	0.999	**0.072**
shape | material fixed	11	0.091	0.999	**0.998**	1.000	**1.000**	0.999	**1.000**
shape | color fixed	11	0.091	0.999	**0.991**	1.000	**0.970**	0.999	**0.969**

Once solved, the optimal constituent embeddings $x_c^\star$ are recovered directly by inverting the change of variables:

$$\alpha_c^\star = \beta_c^\star / w_c^\star, \qquad x_c^\star \;=\; \hat{\mu}_c + \hat{U}_c \alpha_c^\star \;\in\; \mathcal{B}_c. \tag{7}$$

This operation yields a different embedding for each constituent concept, placed at the exact location within its compact support region determined by the optimization, thereby isolating the scene-specific attributes of that concept. Figure 3 visualizes a dual-object sample and its two decomposed components.

4.3 Attribute Binding

The SOCP objective, Equation (6), minimizes pure ℓ_2 reconstruction error without explicit attribute-binding supervision. In principle, $\alpha_c^\star$ could land anywhere within its support region $\mathcal{B}_c$. Yet, empirically, the recovered components $x_c^\star$ bind attributes to the correct object with remarkably high accuracy (See Table 1). Why does an objective that only seeks to reconstruct e bind "blue" to the sphere rather than the cube, when both concepts support that attribute?

We hypothesize that while attributes like color or material exhibit coarse global directions across concepts, they also possess finer granularity conditioned on object identity. We validate this on single-object embeddings z_c by subtracting the concept mean, $\tilde{z}_c \;=\; z_c - \mu_c$, which centers every blob at the origin, removing macro object identity (*anonymization*). Linear probes on the anonymized embeddings $\tilde{z}_c$ for a single attribute demonstrate why a global direction for attributes is not sufficient. Across the entire anonymized space, shape probes fail completely ($\approx 7\%$, Table 2), confirming that global identity is removed. Yet, even in this anonymized space, restricting evaluation to a single attribute value allows us to recover object identity almost perfectly ($\approx 99.7\%$, Table 2). We repeat this process visually when fixing two attributes (Fig. 9), validating that attributes are intrinsically conditioned on the host concept:

$$(\mathbf{a} \mid c_i) \;\neq\; (\mathbf{a} \mid c_j) \quad \text{for } i \neq j. \tag{8}$$

Since anonymized displacements retain object-specific structures, a global ℓ_2 reconstruction fit, Equation (6), can only reach minimal error when attributes are assigned to their correct concepts.

5 Analysis

To demonstrate the generality of our framework, evaluating it across diverse architectures and pre-training models: vision-language, vision-only and text-only encoders. Since evaluating attribute binding needs per-object labels, we build two controlled datasets (the specific implementation details are in Section E), one visual and one textual (Fig. 8). Moreover, we demonstrate the applicability also to scenes with three objects, and to natural and generated objects (cat and dog classes).

Decomposition performance comparison. We compare our decomposition method against established latent composition baselines in Table 1. Note that comparing against Sparse Autoencoders (SAEs) and dictionary learning methods is not feasible as they do not perform multi-object binding. While additive baselines can often isolate individual attributes in simple cases, they degrade rapidly

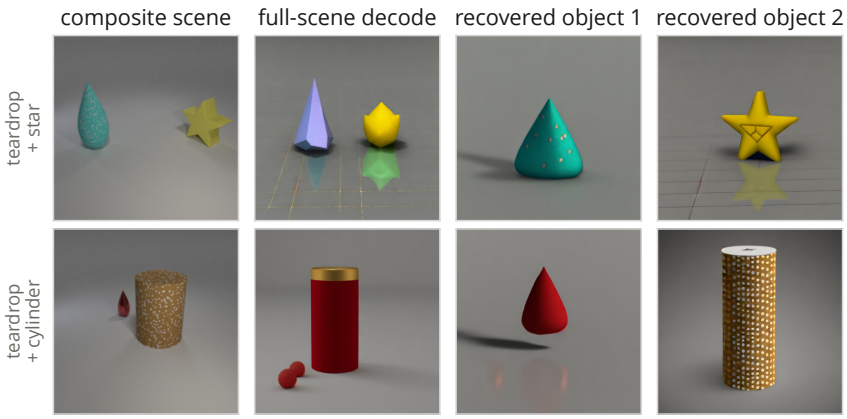

Figure 3: **Dual-object Decomposition.** Even when the composite's decode loses information, the recovered components show each object with its own attributes. More examples are in Section D.

when required to bind multiple attributes simultaneously to their respective objects. Implementation details on this controlled environment are in Section F.

Binding performance across encoders. We validate that attribute binding generally holds across different encoders and in different domains (image and text). In Fig. 5, we report the accuracy of several cutting-edge encoders in predicting the correct attributes from the decomposed samples. We also report performance when including the "and" blob, showing consistent improvement across all encoders and all attributes, where there is a higher gain in the text domain. We observe that binding performance correlates with the blob hierarchy (Fig. 10), where across all embedders material is bound better than color. Another observation is that DINOv3 outperforms all other encoders in correctly binding position (left/right), supporting the common belief that self-supervised visual features localize better than contrastive vision-language ones (El Banani et al., 2024). For a visual impression of the decomposition, see Fig. 3; additional examples are provided in Section D.

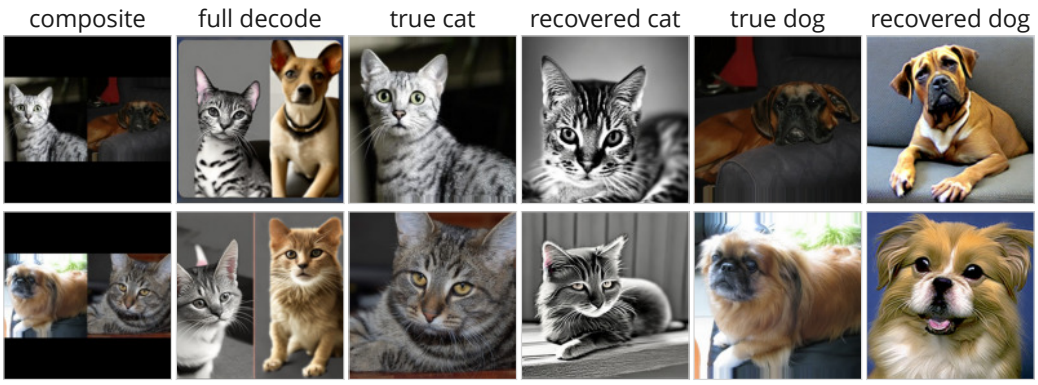

Figure 4: **Decomposing real images.** For real images as well, the decomposition usually attaches the correct attributes to the objects, such as breed, color and pose. More examples, including generated unified scenes, are in Figs. 22 to 25.

Real image decomposition. We validate our decomposition on real images by constructing composite images formed by concatenating ImageNet samples from 10 classes: dog and cat, 5 breed classes each. In Fig. 4, we display composites alongside generations synthesized from both the combined representation and the individual decomposed components, next to the true single-animal crops. Visually, the decomposition yields promising results where attributes like breed, color, pose and surroundings are bound correctly. Implementation details and additional examples are in Section H.

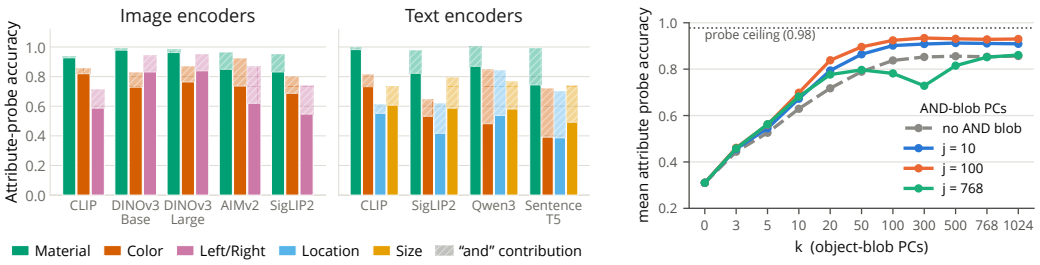

Figure 5: **Decomposition across foundation models, and "and" blob contribution.** Left: Accuracy of the attribute probes on the recovered components, per attribute, for various image encoders and text encoders. Right: Mean attribute-probe accuracy across ranks of the concept blob (k) and the "and" blob (j), for DINOv3 ViT-L/16.

Rank and separability of concept blobs. We study the optimal rank of both the concept blobs and the "and" blob. In Fig. 5 (right), we demonstrate this for DINOv3, revealing that the concept-blobs saturate at a dimension around 100 (vs. 1024 for DINOv3 ViT-L/16). Surprisingly, the "and" blob does not behave the same way: its contribution peaks at a rank of about 100 and probe accuracy degrades beyond it, so a richer relational blob is not a better one. In addition to their low intrinsic dimensionality, these structures exhibit strict spatial separation: as object concepts are distinct, their corresponding blobs are mutually exclusive; testing all object pairs, no sample of one object falls inside the blob of another (Fig. 6). Full analysis for all encoders is in Section E.5.

Beyond two objects. As the number of objects in a scene increases, decomposition becomes inherently more complex due to a higher chance of occlusions. We validate our binding accuracy on three-object scenes in Fig. 19, while binding performance degrades gracefully with larger amount of objects, it remains remarkably strong: for instance, DINOv3 recovers the material of each of the three objects with over 90% accuracy. Implementation details are in Section G.

Dark matter. We render triplets of a two-object composite and the two single-object renders it was built from (Section E.1, Fig. 8). Even when the ground-truth concepts are known in advance, the best weighted sum of equation 5 reaches a cosine of only 0.839–0.986 with the composite, a gap we refer to as representation *dark matter* (Section B). The residuals across the dual-concept dataset are systematic rather than random noise, and are recovered as an additive concept, the "and" blob.

6 Conclusion

In this paper, we introduced *Concept-Blobs*, a novel probabilistic geometric framework that models concepts in foundation-model representations as compact Mahalanobis support regions rather than rigid one-dimensional vectors. By modeling multi-concept compositions through a scaled Minkowski sum, our framework reveals two fundamental structural properties of latent representation spaces. First, introducing a dedicated relational "and" blob ($\mathcal{B}_{\text{and}}$) accounts for inter-concept interactions, resolving the residual reconstruction gap previously identified as representation *dark matter*. Second, attribute binding emerges naturally from high-dimensional geometry without requiring external constraints or binding losses, as global attribute directions partition into object-specific sub-directions. Across nine frozen encoders in both visual and textual modalities, our empirical findings demonstrate that our framework consistently preserves compositional fidelity and attribute-object associations, showing strong generalization to real images.

Limitations. Calibrating explicit covariance matrices Σ_c and principal directions U_c requires clean, labeled samples for each concept, which can be non-trivial to curate for open-world, highly abstract concepts. Moreover, while our convex Minkowski recovery scales effectively to three-object scenes, extending it to highly cluttered multi-object scenes introduces combinatorial complexity in identifying constituent concept subsets. This complexity can be mitigated by first detecting and classifying candidate objects, tasks where modern vision models achieve high accuracy, and then applying our approach to bind their respective attributes.

REPRODUCIBILITY STATEMENT

To ensure complete reproducibility, all implementation details are in the Appendix. All Minkowski sum decompositions are based on fully available package implementation. Upon acceptance, our complete codebase will be publicly released.

LLM USAGE

Large language models were used for writing assistance, such as improving language clarity, correcting grammar, generating illustrations, and for code generation, limited to implementing specified functions or refactoring existing code according to explicit instructions. All experimental design choices, analyses, interpretations, and conclusions were made by the authors. The authors are fully responsible for all content, claims, and conclusions presented in this work.

REFERENCES

Reza Abbasi, Mohammad Hossein Rohban, and Mahdieh Soleymani Baghshah. Deciphering the role of representation disentanglement: Investigating compositional generalization in CLIP models. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 35–50, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-73024-5.

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. In *Advances in Neural Information Processing Systems*, 2024.

Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. LEACE: Perfect linear concept erasure in closed form. In *Advances in Neural Information Processing Systems*, 2023.

Davide Berasi, Matteo Farina, Massimiliano Mancini, Elisa Ricci, and Nicola Strisciuglio. Not only text: Exploring compositionality of visual representations in vision-language models. In *CVPR*, 2025.

Roy Betser, Meir Yossef Levi, and Guy Gilboa. Whitened CLIP as a likelihood surrogate of images and captions. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 4069–4095. PMLR, 13–19 Jul 2025.

Roy Betser, Eyal Gofer, Meir Yossef Levi, and Guy Gilboa. InfoNCE induces gaussian distribution. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=BlSH7gNQSq.

Usha Bhalla, Alex Oesterling, Suraj Srinivas, Flavio P. Calmon, and Himabindu Lakkaraju. Interpreting clip with sparse linear concept embeddings (splice). In *NeurIPS*, 2024.

Yichao Cai, Zhen Zhang, Yuhang Liu, and Javen Qinfeng Shi. The geometric mechanics of contrastive representation learning: Alignment potentials, entropic dispersion, and cross-modal divergence. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=X9VtjdBlQw.

Declan Campbell, Sunayana Rane, Tyler Giallanza, Nicolò De Sabbata, Kia Ghods, Amogh Joshi, Alexander Ku, Steven M. Frankland, Thomas L. Griffiths, Jonathan D. Cohen, and Taylor Webb. Understanding the limits of vision language models through the lens of the binding problem. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 113436–113460. Curran Associates, Inc., 2024. doi: 10.52202/079017-3604. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/cdcc6d47c1627350014a3076112ab824-Paper-Conference.pdf.

Andrew Draganov, Sharvaree Vadgama, Sebastian Damrich, Jan Niklas Böhm, Lucas Maes, Dmitry Kobak, and Erik J Bekkers. On the importance of embedding norms in self-supervised learning. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=L8YhRgEcZp.

Mohamed El Banani, Amit Raj, Kevis-Kokitsi Maninis, Abhishek Kar, Yuanzhen Li, Michael Rubinstein, Deqing Sun, Leonidas Guibas, Justin Johnson, and Varun Jampani. Probing the 3D awareness of visual foundation models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. Toy models of superposition. *arXiv preprint arXiv:2209.10652*, 2022.

Joshua Engels, Logan Riggs Smith, and Max Tegmark. Decomposing the dark matter of sparse autoencoders. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=sXq3Wb3vef.

Thomas Fel, Matthew Kowal, Mozes Jacobs, et al. Structuring sparsity: Block-sparse featurizers capture visual concept manifolds. *arXiv preprint arXiv:2606.25234*, 2026.

Ehud Gordon, Meir Yossef Levi, and Guy Gilboa. Scocca: Multi-modal sparse concept decomposition via canonical correlation analysis. *arXiv preprint arXiv:2603.13884*, 2026.

Klaus Greff, Sjoerd van Steenkiste, and Jürgen Schmidhuber. On the binding problem in artificial neural networks. *arXiv preprint arXiv:2012.05208*, 2020.

Bijay Gurung, David T. Hoffmann, and Thomas Brox. Common data properties limit object-attribute binding in CLIP. In *DAGM German Conference on Pattern Recognition (GCPR)*, pp. 353–368, 2025.

Robert Huben, Hoagy Cunningham, Logan Smith, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *International Conference on Learning Representations*, 2024.

Yibo Jiang, Goutham Rajendran, Pradeep Ravikumar, Bryon Aragam, and Victor Veitch. On the origins of linear representations in large language models. In *ICML*, 2024.

Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2901–2910, 2017.

Raphi Kang, Yue Song, Georgia Gkioxari, and Pietro Perona. Is CLIP ideal? No. Can we fix it? Yes! In *ICCV*, 2025.

Darina Koishigarina, Arnas Uselis, and Seong Joon Oh. Clip behaves like a bag-of-words model cross-modally but not uni-modally. In *International Conference on Learning Representations*, 2026.

Meir Yossef Levi and Guy Gilboa. The double-ellipsoid geometry of CLIP. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 33999–34019. PMLR, 13–19 Jul 2025.

Martha Lewis, Nihal Nayak, Peilin Yu, Jack Merullo, Qinan Yu, Stephen Bach, and Ellie Pavlick. Does clip bind concepts? probing compositionality in large image models. In *Findings of the Association for Computational Linguistics: EACL 2024*, pp. 1487–1500, 2024.

Yihao Li, Saeed Salehi, Lyle Ungar, and Konrad Kording. Does object binding naturally emerge in large pretrained vision transformers? In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=5BS6gBb4yP.

Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in neural information processing systems*, 35:17612–17625, 2022.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *NeurIPS*, 2013.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. In *ICML*, 2024.

Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *ICCV*, 2021.

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.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Simon Schrodi, David T. Hoffmann, Max Argus, Volker Fischer, and Thomas Brox. Two effects, one trigger: On the modality gap, object bias, and information imbalance in contrastive vision-language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=uAFHCZRmXk.

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.

Ziwei Su, Junyu Ren, and Victor Veitch. Optimization dynamics imprint semantic specificity in contrastive embedding norms. *arXiv preprint arXiv:2606.30625*, 2026.

Yingtian Tang, Yutaro Yamada, Yoyo Zhang, and Ilker Yildirim. When are lemons purple? the concept association bias of vision-language models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 14333–14348, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.886. URL https://aclanthology.org/2023.emnlp-main.886/.

Chen Tasker, Roy Betser, Eyal Gofer, Meir Yossef Levi, and Guy Gilboa. The universal normal embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 32015–32025, June 2026.

Matthew Trager, Pramuditha Perera, Luca Zancato, Alessandro Achille, Parminder Bhatia, and Stefano Soatto. Linear spaces of meanings: Compositional structures in vision-language models. In *ICCV*, 2023.

Anne M. Treisman and Garry Gelade. A feature-integration theory of attention. *Cognitive Psychology*, 12(1):97–136, 1980.

Arnas Uselis, Andrea Dittadi, and Seong Joon Oh. Does data scaling lead to visual compositional generalization? In *ICML*, 2025.

Arnas Uselis, Andrea Dittadi, and Seong Joon Oh. Compositional generalization requires linear, orthogonal representations in vision embedding models. In *Forty-third International Conference on Machine Learning*, 2026a. URL https://openreview.net/forum?id=AQZZWVp6XA.

Arnas Uselis, Darina Koishigarina, and Seong Joon Oh. How can embedding models bind concepts? In *Forty-third International Conference on Machine Learning*, 2026b. URL https://openreview.net/forum?id=lFkGJ60bGq.

Zijie Yu, Gaowen Liu, Ramana Rao Kompella, Philip S Yu, and Yue Song. The hyperspherical geometry of clip latent space: A semantic mixture model. *arXiv preprint arXiv:2607.13660*, 2026.

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? In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=KRLUvxh8uaX.

Qi Zhang, Yuxu Chen, Lei Deng, and Lili Shen. ABE-CLIP: Training-free attribute binding enhancement for compositional image-text matching. *arXiv preprint arXiv:2512.17178*, 2025.

Appendix

A Blob overlap across objects

We quantify the disjointness of the concept blobs by asking how much of one object's samples fall inside another object's support region:

$$\mathrm{ovl}(a, b) \;=\; \frac{1}{|\mathcal{D}_b|} \sum_{z \in \mathcal{D}_b} \mathbf{1}\big[\, z \in \mathcal{B}_a \,\big], \tag{9}$$

with $\mathcal{B}_a$ the blob of equation 1 and $\mathcal{D}_b$ the held-out samples of object b. Each object's $1{,}000$ samples are split $80/20$, and μ_c, U_c, Λ_c and τ_c are estimated on the training part alone, so every cell is evaluated on samples the region has never seen. Diagonal cells measure generalization: at $k = 200$ they average 0.945 (0.925–0.970), close to the 0.95 target. Off-diagonal cells measure disjointness, and they are exactly zero across all eleven objects (Fig. 6): no held-out sample of any object ever falls inside any other object's support region.

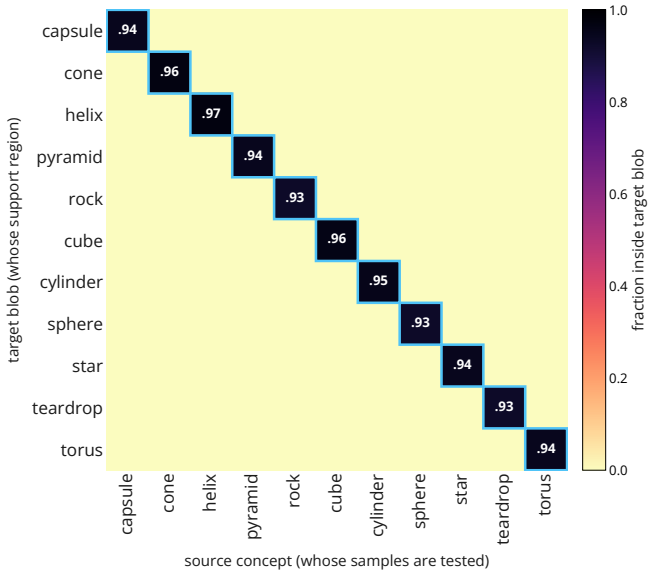

Figure 6: **Concept blobs are disjoint on held-out samples.** Cell $(\text{target}, \text{source})$ is the fraction of the *source* object's *held-out* single-object embeddings falling inside the *target* object's 95th-percentile support region (equation 9), over the eleven objects of Section E.1. The diagonal sits at 0.945 on average, close to the 0.95 the calibration targets. Every off-diagonal cell is exactly zero.

B Residual structure

This appendix collects the evidence behind the "and" blob. Table 3 measures how much of a composite embedding additive composition leaves unexplained: for every calibration composite we solve equation 5 with its own ground-truth constituents and report the cosine between the composite and the best weighted sum. The gap appears in all nine encoders and in both modalities (cosine 0.839–0.986), so it is neither an artefact of one model nor of estimating the constituents. Figure 7 shows that what is left over is not noise: a single draw from $\mathcal{B}_{\text{and}}$, the blob fitted to these residuals, added to a single-concept embedding and decoded with unCLIP, puts a second object in the scene while the base object keeps its identity and attributes. The residual therefore carries the "another object is present" part of the composite, which is why we model it as a concept blob of its own.

Table 3: **What additive composition leaves behind.** Cosine between a composite embedding and the closest weighted sum of its two *true* constituents, equation 5 solved per composite over the 10,000 calibration composites of each domain. Even with the ground-truth constituents in hand, no encoder in either modality closes the gap. The residual left by this fit is what $\mathcal{B}_{\text{and}}$ is built from.

Image encoder	$\cos(e,\ w_1x_1 + w_2x_2)$
CLIP ViT-L/14	0.949 ± 0.016
DINOv3 ViT-B/16	0.904 ± 0.031
DINOv3 ViT-L/16	0.930 ± 0.024
AIMv2 L/14	0.986 ± 0.004
SigLIP2 B/16	0.954 ± 0.011

Text encoder	$\cos(e,\ w_1x_1 + w_2x_2)$
CLIP ViT-L/14	0.895 ± 0.031
SigLIP2 B/16	0.839 ± 0.045
Qwen3-Emb 0.6B	0.953 ± 0.010
Sentence-T5 base	0.977 ± 0.008



Figure 7: **Adding a draw from the "and" blob puts a second object in the scene.** For each concept: the source render, the unCLIP decode of its embedding x, and the decode of $x + x_{\text{and}}$ with x_{and} a single random draw from $\mathcal{B}_{\text{and}}$. Both decodes of a row share a seed and a starting latent, so the only difference between them is x_{and}. The base object keeps its identity and attributes in every case, and a second object appears, most often a smaller copy of the base concept rather than a different one.

C THE SUPPORT REGION AS A SECOND-ORDER CONE

Fitted in the natural variables (w_c, α_c), the composition model of equation 3 is bilinear: the objective contains the product $w_c U_c \alpha_c$, so it is not jointly convex and alternating minimization returns no certificate. Introducing

$$\beta_c \;:=\; w_c\,\alpha_c \tag{10}$$

as the decision variable in place of α_c removes this. Geometrically β_c is the displacement concept c contributes to $\hat{e}$, measured in the composite's scale, where α_c is its position inside its own blob; for $w_c > 0$ the two carry the same information. The model becomes linear in all unknowns,

$$\hat{e} \;=\; \sum_{c \in \mathcal{S}'} \big(w_c\,\mu_c + U_c \beta_c\big), \tag{11}$$

and the support region becomes a constraint coupling β_c to its own weight.

Proposition 1 (Exact convex form of the support region)**.** *For $w_c > 0$ and $\beta_c = w_c \alpha_c$,*

$$\|\Lambda_c^{-1/2}\alpha_c\| \le \tau_c \quad\Longleftrightarrow\quad \|\Lambda_c^{-1/2}\beta_c\| \le \tau_c\, w_c,$$

and the right-hand set is jointly convex in (β_c, w_c).

Proof. Substituting $\alpha_c = \beta_c / w_c$ and multiplying by $w_c > 0$ gives the equivalence. The right-hand constraint is the perspective of the norm $\|\Lambda_c^{-1/2} \cdot\|$, i.e. the statement that $(\tau_c w_c,\ \Lambda_c^{-1/2}\beta_c)$ lies in the second-order cone; both are jointly convex in (β_c, w_c). □

The reparametrization therefore costs nothing: it enforces exactly the intended geometric constraint while making the feasible set convex, with w_c entering only linearly to scale the cone. Recovering the per-concept point after the solve is one division, $\alpha_c^\star = \beta_c^\star / w_c^\star$ (equation 7).

The fitted weights. We use box bounds $w_{\min} = 0.05$ and $w_{\max} = 3$, and solve with a conic solver (CLARABEL, falling back to SCS). Eigenvalues are floored at $10^{-6}\lambda_1$ before $\Lambda_c^{-1/2}$ is formed. The bounds are never active. Table 4 reports the weights the program returns for each encoder, at the operating point it is read at in Fig. 5. The object weights concentrate near 0.5 in every encoder and both modalities. Their sum stays close to 1. Its standard deviation is also markedly smaller than that of the individual weights, so the fit trades weight between the two objects while holding their total nearly fixed: where w_{obj_1} comes out lower, w_{obj_2} comes out correspondingly higher. The "and"-blob weight sits near 1 instead, so $\mathcal{B}_{\text{and}}$ enters the reconstruction at roughly full strength rather than as a small correction.

Table 4: **Fitted weights of the convex program.** Mean ± standard deviation of the weights returned by equation 6 over the evaluation composites. The first column pools w_a and w_b since the two objects of a composite are exchangeable. The second is their sum within a single solve, which stays close to 1 in every encoder, as does the weight on the "and" blob.

Image encoder	w_{obj}	$w_{obj_1} + w_{obj_2}$	w_{and}
CLIP ViT-L/14	0.513 ± 0.064	1.026 ± 0.039	1.015 ± 0.158
DINOv3 ViT-B/16	0.548 ± 0.067	1.096 ± 0.045	1.103 ± 0.176
DINOv3 ViT-L/16	0.542 ± 0.085	1.084 ± 0.036	1.079 ± 0.135
AIMv2 L/14	0.494 ± 0.011	0.988 ± 0.005	1.536 ± 0.079
SigLIP2 B/16	0.493 ± 0.056	0.985 ± 0.017	0.968 ± 0.172

Text encoder	w_{obj}	$w_{obj_1} + w_{obj_2}$	w_{and}
CLIP ViT-L/14	0.483 ± 0.059	0.967 ± 0.062	1.045 ± 0.087
SigLIP2 B/16	0.398 ± 0.067	0.795 ± 0.058	1.017 ± 0.061
Qwen3-Emb 0.6B	0.556 ± 0.035	1.113 ± 0.032	1.330 ± 0.076
Sentence-T5 base	0.395 ± 0.061	0.790 ± 0.050	0.959 ± 0.073

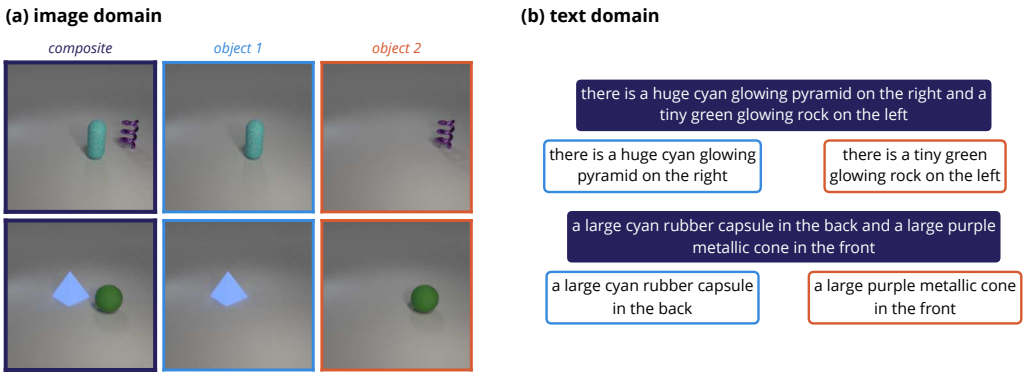

Figure 8: **The two controlled domains.** **(a)** Image domain: each composite scene beside the two single-object renders it was built from. **(b)** Text domain: each composite caption above the two single-object captions that describe each of its objects alone. These are the triples that equation 5 is fitted on.

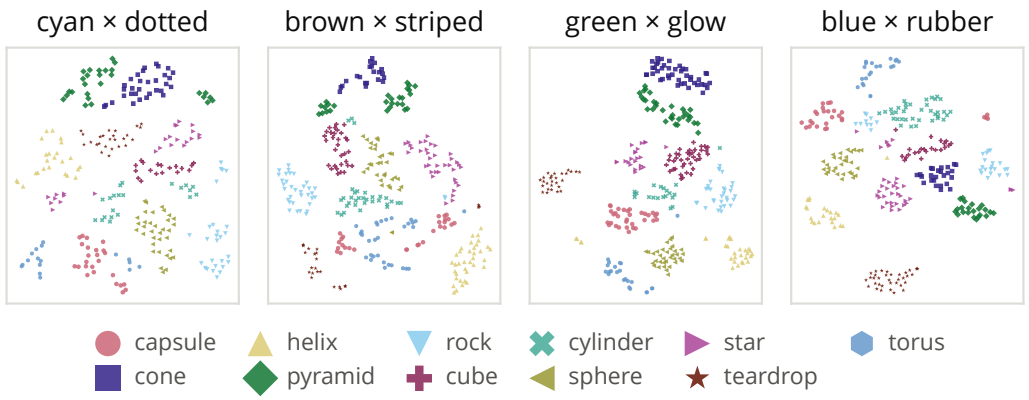

Figure 9: **Conditioning makes concept structure visible.** t-SNE of anonymized CLIP embeddings, colored and marked by shape, for four randomly chosen (color × material) cells. Anonymization leaves no global shape signal (Table 2), yet within each cell the shapes cluster cleanly. We believe this behavior allows binding of attributes to specific objects.

D DECODED COMPONENTS

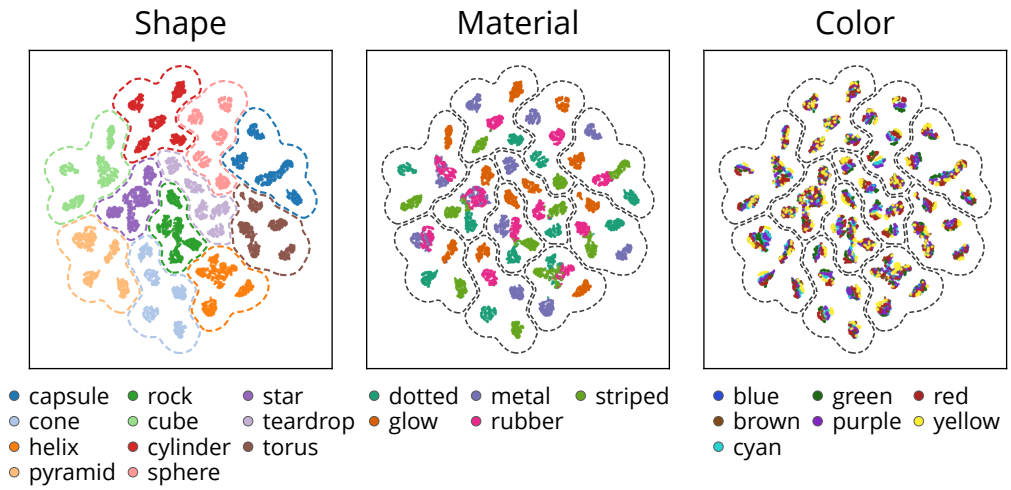

Figure 10: **Attributes are organized hierarchically inside the blob.** t-SNE of single-object embeddings (DINOv3 ViT-L/16), colored by shape, material, and color. Outlines mark the region of each shape and are repeated on every panel. Clear hierarchy emerges: object identity (shape), then material partition, then color within each part. Size and position vary within cells and are invisible at this scale.

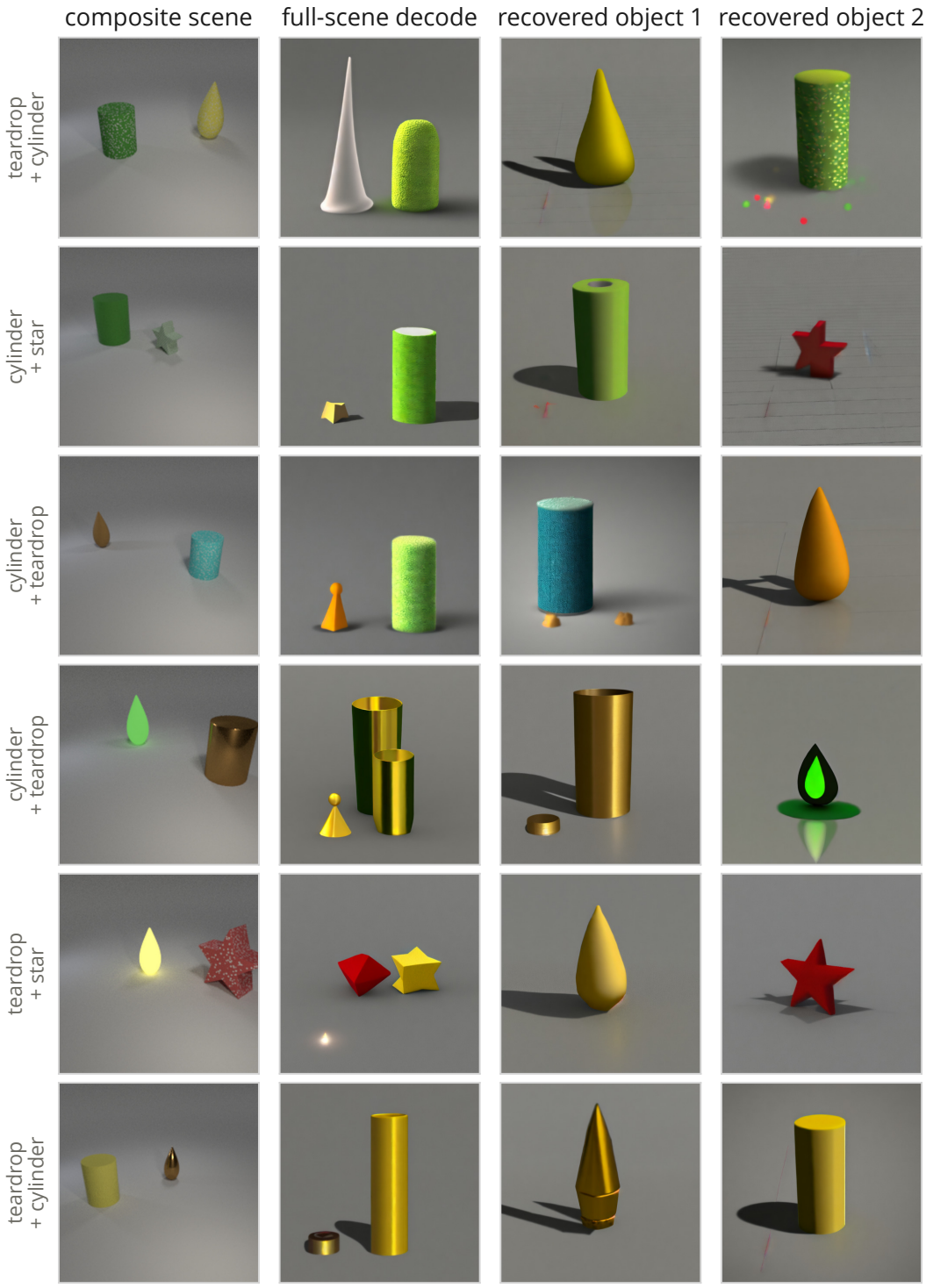

Figure 11: **Decoded components, 1 of 3.**

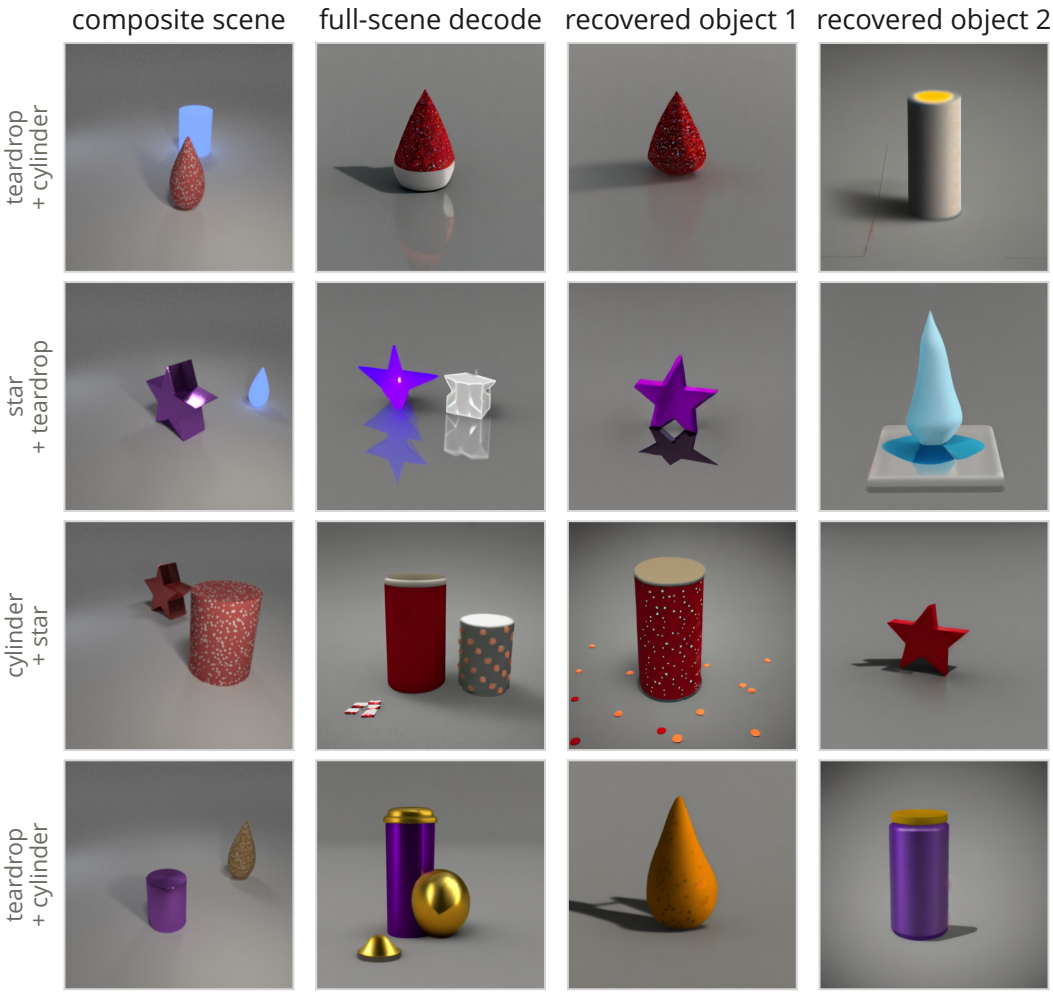

Figure 12: **Decoded components, 2 of 3.**

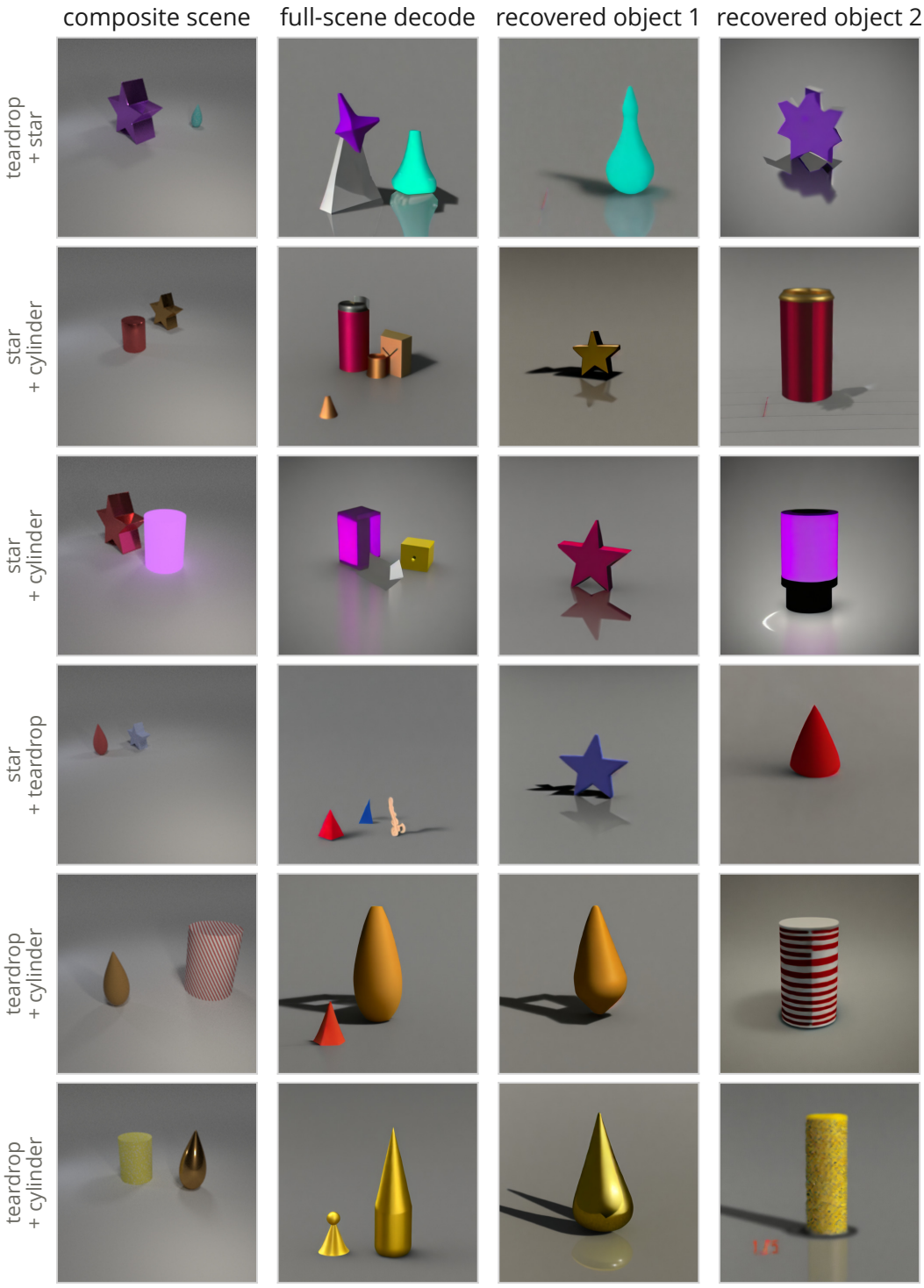

Figure 13: **Decoded components, 3 of 3.**

E EXPERIMENTAL DETAILS

E.1 IMAGE DOMAIN

We render CLEVR (Johnson et al., 2017) scenes over 11 shape concepts: capsule, cone, helix, pyramid, rock, cube, cylinder, sphere, star, teardrop and torus. Each object independently receives one of 7 colors (red, blue, green, brown, purple, cyan, yellow) and one of 5 materials (rubber, metal, dotted, striped, glow; Fig. 15). Its size, the object's radius in scene units, is drawn uniformly from $[0.65, 1.55]$, and its center is drawn uniformly over the $[-3, 3]^2$ ground plane, from which we derive a binary left/right class. Attributes are sampled independently of shape, so no attribute is predictable from concept identity alone.

Singles. 1,000 renders per shape (11,000 total) with independently randomised attributes, for all eleven shapes; Fig. 14 shows eight random singles per shape. These calibrate the concept blobs (Section 4.1) and fit the attribute probes (Section E.4).

"And"-blob calibration composites. 10,000 scenes of two independently sampled objects with random positions and sizes, each accompanied by the two single-object renders it was built from (Fig. 8(a)). Only eight shapes appear in them (sphere, cone, pyramid, torus, capsule, helix, rock, cube). The residuals of equation 5 over these scenes are the samples $\mathcal{B}_{\text{and}}$ is built from.

Evaluation composites. A separate render set, never used in calibrating anything, on three held-out shapes (cylinder, teardrop, star), scored in Table 1. Because the "and" blob is calibrated on residuals pooled over the eight calibration shapes, any gain it delivers on the held-out shapes is a gain on concept pairs it has never encountered.

E.2 TEXT DOMAIN

The text set deliberately mirrors the image set: the same 11 shapes, 7 colors and 5 materials, and the same eight-versus-three shape split. Captions are built by wrapping a noun phrase `[size] [color] [material] [shape]` in one of 10 carrier templates ("a photo of {NP}", "there is {NP}", ...) and appending a location phrase. Two factors differ from the image domain by construction: size takes 4 lexical values (tiny/small/large/huge) rather than a continuous scale, and location is a nuisance phrase drawn from 6 strings rather than a pixel-derived class.

Singles. 1,000 captions per shape (11,000 total); Fig. 16 shows how a caption is built and ten random examples.

"And"-blob calibration composites. 10,000 captions, each joining two noun phrases with "and" inside a single template and accompanied by matched single-object captions for both halves (Fig. 8(b)), over the eight calibration shapes.

Evaluation composites. 1,000 captions over the three held-out shapes (Table 1).

E.3 ENCODERS

Every encoder is frozen and used only as a feature extractor. In the image domain we embed each render with five of them: CLIP ViT-L/14 (`openai/clip-vit-large-patch14`, 768-d), DINOv3 ViT-B/16 (768-d) and ViT-L/16 (1024-d), AIMv2 L/14 (1024-d), and SigLIP2 B/16 (768-d). In the text domain we embed each caption with four: the text encoders of CLIP ViT-L/14 (768-d) and SigLIP2 B/16 (768-d), Qwen3-Embedding-0.6B (1024-d), and Sentence-T5 base (768-d). The set spans contrastive vision–language models, purely visual self-supervision, generative pre-training and dedicated sentence embedders, so that nothing we report rests on a single training objective. We take the pooled vector each model exposes, the projected feature for the contrastive models and the pooled token for the rest. Embeddings are used raw: unnormalised and not globally mean-centred. From this point the pipeline is encoder-agnostic: the blobs of equation 1, the program of equation 6, the decomposition of equation 7 and the probes below consume the vectors as they are, with no per-encoder tuning beyond the subspace ranks that are swept for every model alike.



Figure 14: **Single-object renders, eight random samples per object.** One row per object over the eleven objects of Section E.1, drawn uniformly at random from that object's 1,000 single-object renders. Within a row only the object identity is held fixed; color (one of 7), material (one of 5), size and position are sampled independently.

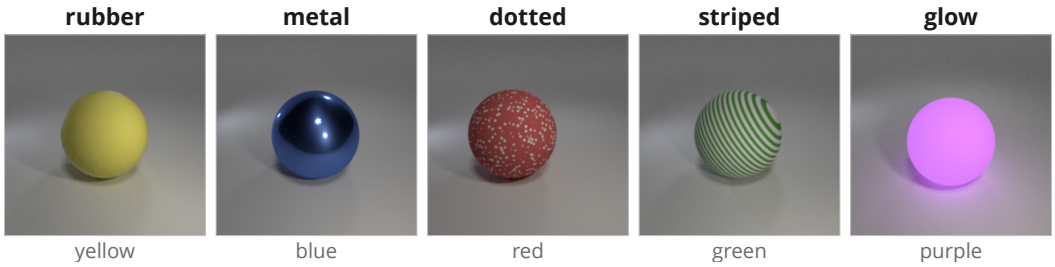

Figure 15: **The five materials, each in a different color.** One single-object render per material (rubber, metal, dotted, striped, glow), with the shape fixed to a sphere so that material and color are the only factors that change, cropped around the object. Every render in Fig. 14 draws its material and color independently from these sets.

carrier	there is {NP} · a photo of {NP} · we can see {NP} · an image of {NP} a picture of {NP} · a close-up of {NP} · a 3D render of {NP} the image shows {NP} · a scene containing {NP} · {NP}
{NP}	a [size] [color] [material] [shape]
size	huge · large · small · tiny
color	blue · brown · cyan · green · purple · red · yellow
material	dotted · glowing · metallic · rubber · striped
shape	capsule · cone · cube · cylinder · helix · pyramid · rock sphere · star · teardrop · torus
location	in the back · in the center · in the front · on the left on the right · (none)
examples	a 3D render of a small blue glowing sphere in the front a close-up of a huge brown metallic helix in the center a scene containing a tiny green glowing cone in the front we can see a small brown metallic cone in the back a close-up of a tiny cyan metallic star on the right an image of a tiny purple metallic star on the right a picture of a tiny red metallic sphere in the front a picture of a tiny cyan rubber helix on the right there is a large purple glowing helix in the back there is a tiny green metallic rock in the back

Figure 16: **How a single-object caption is built. Top:** the 10 carrier templates, the noun-phrase skeleton, the vocabulary each slot draws from, and the 6 location phrases (one of them empty). **Bottom:** ten captions drawn uniformly at random from the 11,000 singles. Words are colored by the factor they realize; uncolored text is the carrier template, which carries no factor.

E.4 Attribute probes and their ceilings

Probes are the measurement instrument for decomposition, so they are fitted *only* on genuine single-object embeddings and never see a composite or a recovered component. For each attribute we fit one-vs-rest binary logistic regressions (a single linear layer per class with `BCEWithLogitsLoss`, fitted independently) and predict by the argmax over the per-class logits, on a random $90/10$ instance-level split of the singles with identical hyperparameters in both domains (Adam, lr $= 0.05$, weight decay $= 10^{-3}$, 300 epochs). We probe four attributes in the image domain (color, material, size, left/right) and four in text (color, material, size, location).

A recovered component is at best as informative as a genuine single-object embedding, so each probe's accuracy on the *held-out singles* ($1{,}100$ per encoder) is the most any decomposition could recover on that attribute. The ceiling is a property of the encoder, not of the method, and the tables below give the full set.

Image encoder	Color	Material	Left/right
CLIP ViT-L/14	1.000	0.995	0.869
DINOv3 ViT-B/16	0.998	1.000	0.941
DINOv3 ViT-L/16	1.000	1.000	0.933
AIMv2 L/14	1.000	0.999	0.908
SigLIP2 B/16	1.000	0.997	0.888

Text encoder	Color	Material	Size	Location
CLIP ViT-L/14	1.000	1.000	1.000	1.000
SigLIP2 B/16	1.000	1.000	1.000	1.000
Qwen3-Emb 0.6B	1.000	1.000	1.000	1.000
Sentence-T5 base	1.000	1.000	1.000	1.000

Color and material are recoverable from a single-object embedding by every encoder in both domains. Left/right is the exception in the image domain: no encoder reaches 0.95 and CLIP is the weakest at 0.869, which is why an image encoder can appear to lose more of that attribute than it really does when scored against 1. In the text domain every factor is a literal word in the caption and all four encoders read every one of them perfectly. Size is not shown for the image domain in Fig. 5: binned into 10 classes, the probe's ceiling accuracy on size was only ~ 0.3, too uninformative to report.

E.5 Fitting and decomposition

With the blobs calibrated and the probes fitted, the experiment is the same three steps in every domain and for every encoder.

Calibrate the concept blobs. From the single-object embeddings of one encoder we estimate each concept blob: mean μ_c, a rank-k PCA subspace U_c, the directional variances Λ_c and the radius τ_c (Section 4.1). Every concept is calibrated from its own samples alone and never sees a composite.

Build the "and" blob. For every "and"-blob calibration composite we embed the composite and its two single-object inputs with the same encoder and solve the unconstrained two-term fit of equation 5 against that composite's *own* constituent embeddings e_1, e_2, which leaves the residual

$$r \;=\; e \;-\; \big(w_1^\star e_1 + w_2^\star e_2\big). \tag{12}$$

Since e_1 and e_2 are the ground-truth constituents, r is by construction the part of the composite that no weighting of its true constituents can explain. Pooling r over the $10{,}000$ composites gives the sample set the "and" blob is built from, and $\mathcal{B}_{\text{and}}$ is the blob those samples define: mean μ_{and}, rank-j basis U_{and}, directional variances Λ_{and} and radius τ_{and}, obtained exactly as in equation 1. That is the point of treating "and" as a concept.

Fit, decompose and score. For each evaluation composite we solve the program of equation 6 once. The input is the composite embedding e and the identity of the concepts it contains, never the per-object embeddings, which exist only for scoring. The solve returns one weight and one

in-blob point per concept, plus the "and" blob's; equation 7 turns each in-blob point into a recovered single-object embedding $x_c^\star$, which the probes label. A component is correct on an attribute when its probe recovers the value belonging to *that* object. The ranks are swept over $k \in \{0, 3, 5, 10, 20, 50, 100, 300, 500, d\}$ and $j \in \{-1, 0, 3, 5, 10, 20, 50, 100, 300, 500, d\}$, with d the embedding dimension, where $j = -1$ removes the "and" blob entirely and $j = 0$ keeps only its mean. For Fig. 5 each encoder is reported at the (k, j) cell maximising its mean accuracy relative to ceiling over the displayed attributes, and the "without and" part of each bar is the $j = -1$ cell at the same k, so the only thing that changes inside a bar is whether $\mathcal{B}_{\text{and}}$ is in the model. The solver settings are given in Section C.

Figure 17 repeats the (k, j) sweep of Fig. 5 (right) for every encoder. The picture is the same across models: accuracy saturates around k=100, an "and" blob of moderate rank (j=100) helps at every k past the first few.

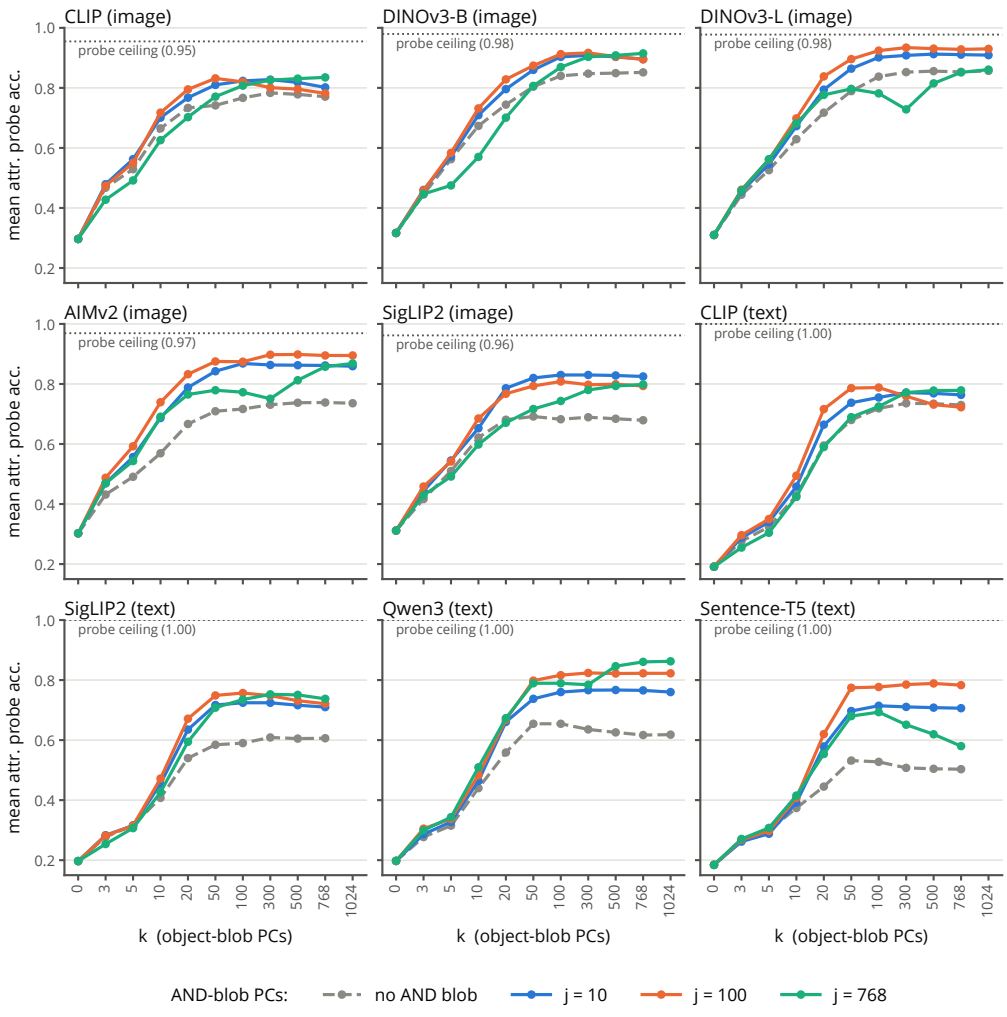

Figure 17: **The (k, j) sweep for every encoder.** Mean attribute-probe accuracy of the recovered components as a function of the concept-blob rank k, without the "and" blob (dashed) and with an "and" blob of rank $j \in \{10, 100, 768\}$.

F Decomposition performance comparison

All three baselines are fit on the same single-object calibration set the concept blobs are calibrated from (Section E.1, Section E.2), per domain and per encoder.

LDE (Trager et al., 2023) represents an object as an anchor plus one vector per attribute. The anchor c_0 is the mean embedding over all singles; each attribute vector u_{a_i} is the difference between the mean embedding of singles carrying value i of that attribute (e.g. "red", "metal") and c_0. An object's embedding is reconstructed as c_0 plus the sum of its own attribute vectors, and the composite estimate $\hat{e} = c_0 + \sum_i u_{a_i} + \sum_i u_{b_i}$ sums the two objects' attribute vectors around a single shared anchor.

GDE (Berasi et al., 2025) carries the same factorization onto the unit sphere. μ is the Fréchet mean of the singles; each attribute vector is first computed in the ambient space exactly as for LDE and then mapped into the tangent space at μ via the Riemannian log map, Log_μ, before being summed per object into v_a and v_b. The composite estimate $\hat{e} = \mathrm{Exp}_\mu(v_a + v_b)$ maps the tangent-space sum back onto the sphere with the exponential map at μ, so composition happens in the tangent space rather than by linear addition in the ambient space.

Single-obj (Uselis et al., 2026b) skips per-attribute decomposition entirely: $\hat{u}_a$ is the mean embedding of the singles sharing object a's exact attribute combination (the same shape, color and material), read directly off the calibration set. The composite estimate is simply $\hat{e} = \hat{u}_a + \hat{u}_b$, the sum of the two objects' empirical means.

For each composition model, we compose a candidate two-object embedding for every assignment of a (color, material) pair to each object, 35×35 candidates in all (object identity is given), and match each composite embedding to its closest candidate by cosine similarity. The comparison is run on CLIP, on the three held-out shapes that no composite used in calibrating the "and" blob (Table 1).

G Generalisation to three-object scenes

Changing to three objects, we simply add a third concept blob to the scaled Minkowski-sum model of equation 3 and refit the "and" blob on the residuals of the resulting three-term fit.

Data. The per-object concept blobs are the same ones calibrated in Section E.1 (image) and Section E.2 (text) from the same single-object data; only the "and" blob needs new data. For the "and" blob we render a three-object counterpart of each domain. In the image domain, the "and" blob is fitted on $10{,}000$ composites with randomised object positions and scales, together with their corresponding $30{,}000$ single-object crops, and evaluated on $1{,}000$ held-out scenes so that the "and" blob is scored out of distribution, exactly the protocol used for two objects. In the text domain we generate $10{,}000$ training and $1{,}000$ test caption quadruples: one composite caption naming three objects and the three single-object captions that describe each of them alone, built from the same templates and factor vocabulary as the two-object set. The shape split is preserved, so the test captions use three shapes never seen in training. Fig. 18 shows examples of both.

Fitting the "and" blob. For each composite we solve the unconstrained three-term problem $\min_w \|e - \sum_{i=1}^{3} w_i e_i\|^2$ against that scene's own single-object embeddings, and fit the "and" blob to the residuals. The fit degrades gracefully relative to two objects: cosine similarity between the weighted sum and the composite falls from 0.949 to 0.923 for CLIP images, and from 0.895 to 0.812 for CLIP text.

Results. Fig. 19 reports attribute-probe accuracy at k operating point, with and without the "and" blob. The finding of the main text carries over: the decomposition still recovers per-object attributes well above chance.

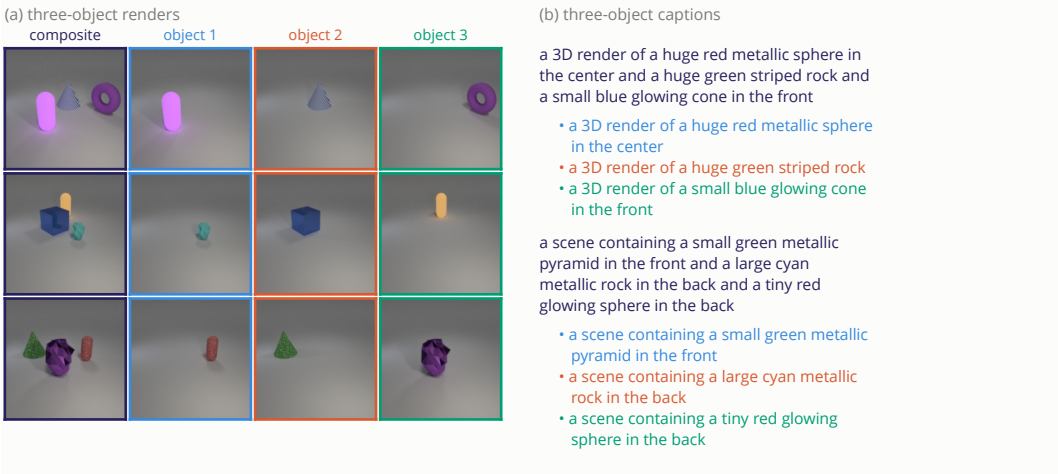

Figure 18: **The three-object data.** **(a)** composite renders beside the three single-object renders the "and" blob is fitted against. **(b)** composite captions above the three single-object captions that describe each of their objects alone, built from the same templates and factor vocabulary as the two-object set. In both domains the composite and its three components form the quadruple the three-term fit is solved on.

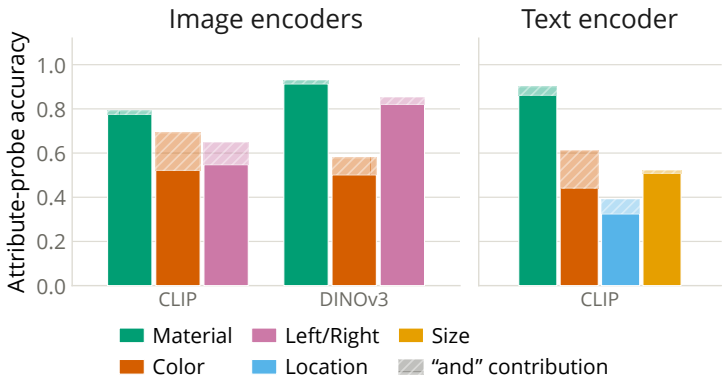

Figure 19: **Generalisation to three-object scenes.** Accuracy of the attribute probes on the recovered components, per attribute, in the image domain and the text domain. Solid bars are the fit without the "and" blob; hatched is what adding $\mathcal{B}_{\text{and}}$ contributes.

H A REAL-IMAGE DOMAIN: IMAGENET CATS AND DOGS

We examine our model on complex, real images from ImageNet classes. Nothing about the model changes: the same concept blobs, the same "and" blob, the same program of equation 6, the same decomposition of Eq. (7).

Concepts and crops. We want a single-object dataset for each concept, *cat* and *dog*, whose members vary along an attribute (here the breed), exactly as the rendered singles of the main text vary in shape, colour and material. ImageNet photographs are not single-object images as they stand: the animal is rarely centred, it may occupy a small part of the frame, and other concepts are often present alongside it. We therefore use ImageNet's built-in human-annotated bounding boxes to isolate one animal per image.

We use ten ImageNet classes, five cat classes (tabby, tiger cat, Persian, Siamese, Egyptian) and five dog breeds (boxer, standard poodle, golden retriever, German shepherd, Pekinese), with 450 images each, 4,496 crops in total. Each image is cut to its bounding box and made square without

anisotropic scaling: the square side is $1.15\max(w_{\text{box}}, h_{\text{box}})$, capped at the image's short side but never below the box itself; the window is grown per axis from real pixels and shifted to stay in frame; only the axis that runs out of image is padded, by edge replication; and the result is resized to 224×224 by a single uniform factor. Fig. 20(a) shows two randomly chosen crops per class.

Composites. Real photographs showing a cat and a dog of these breeds together in one natural scene do exist, but not at the scale, nor with the per-object ground truth, that fitting and scoring a decomposition requires. We therefore construct composites from the crops themselves. A composite pairs a random cat crop with a random dog crop, whole, side by side (448×224), rescaled by one uniform factor to 224×112 and letterboxed in black to 224×224 (Fig. 20(b)). Which animal falls on the left is randomised per composite. We build $10{,}000$ composites for fitting the "and" blob and score on 200 whose cat$\times$dog pairings appear in none of them, so the "and" blob is again evaluated out of distribution.

Ceiling. The probe is a logistic regression over the ten classes, trained *only* on single-animal crops and, critically, with the 385 singles belonging to the test composites excluded ($4{,}111$ training images). The resulting ceiling is 0.8775, and it is asymmetric: 0.995 on the dog side against 0.760 on the cat side. This can be explained by the cat classes being genuinely confusable with one another, tabby, tiger cat and Egyptian cat in particular, whereas the five dog breeds are visually distinct. Cat-versus-dog separability, by contrast, is 1.000: the cat and dog concept blobs are mutually exclusive, even though the breed-level classes within each are not.

Fitting the "and" blob. Exactly as in the main text: for each composite we solve the unconstrained two-term problem $\min_w \|e - (w_1 e_{\text{cat}} + w_2 e_{\text{dog}})\|^2$ against that composite's own single-animal embeddings, and fit $\mathcal{B}_{\text{and}}$ to the pooled residuals. The two-term fit reaches cosine 0.819.

Results. At the best operating point, $k = 50$, $j = 10$, the decomposition recovers the correct ImageNet class for 0.840 of components (95.7% of the probe ceiling) (Fig. 21.) The decomposition recovers per-object identity far above the 0.10 chance rate and close to the ceiling, and the "and" blob pays for itself at every rank tested, contributing $+0.022$ at $k = 50$ and as much as $+0.085$ at $k = 3$, again largest where the concept blobs are given least rank. We also decode the recovered components with unCLIP(Fig. 22)

Generated co-occurrence. To examine natural scenes in which the two breeds appear together, we generate 200 photo-realistic scenes with FLUX.1-schnell, each prompted with a cat and a dog of prescribed breeds, placed independently within the scene. We decompose them with the concept blobs and the "and" blob fitted on the composites above, without refitting. Figs. 23 to 25 show decomposition visualizations.

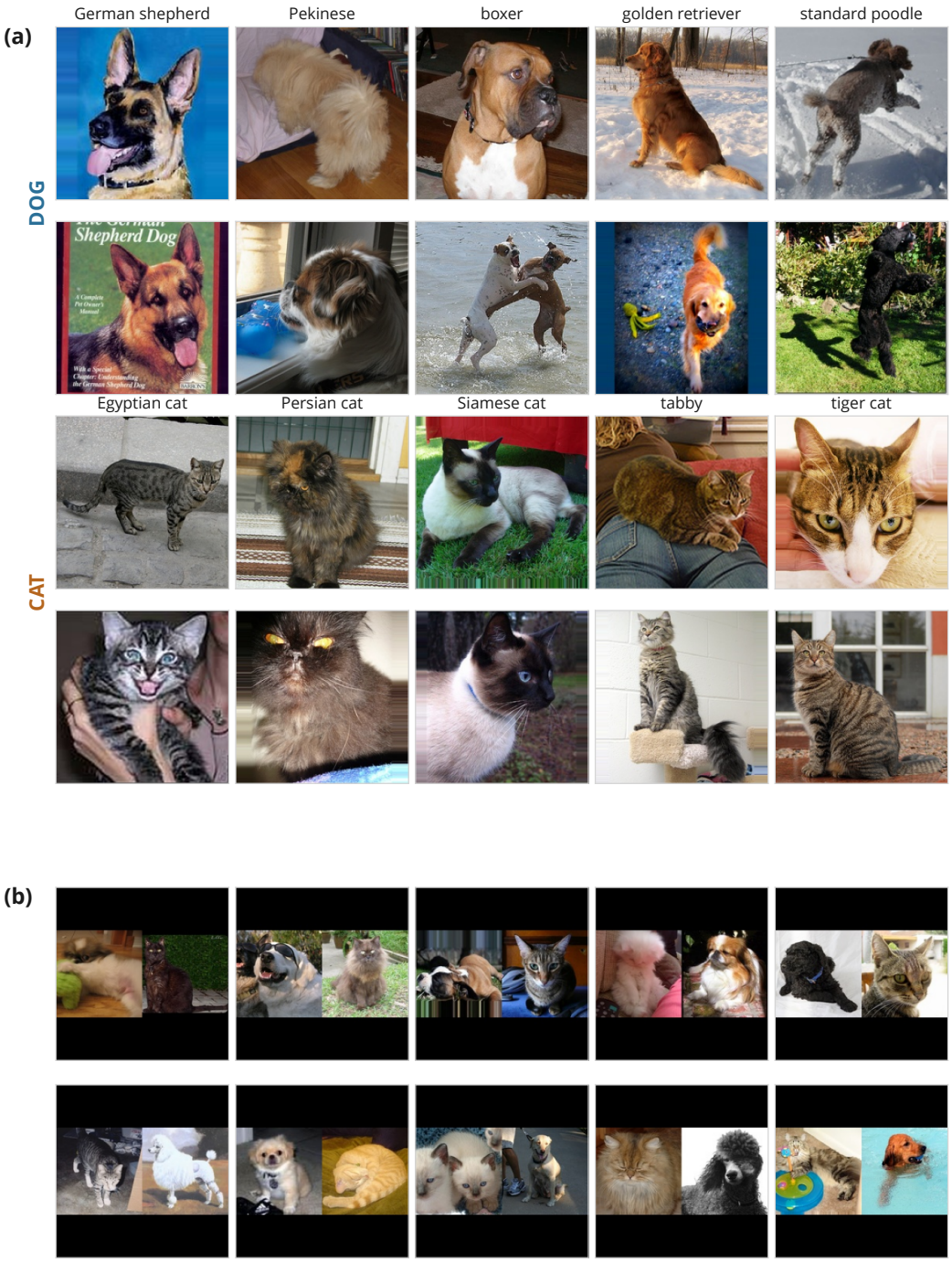

Figure 20: **The ImageNet cat/dog domain. (a)** Two bounding-box crops per class: five dog breeds above five cat classes, 450 images each. **(b)** Ten composites, each pairing a random cat crop with a random dog crop; the side each animal occupies is randomised.

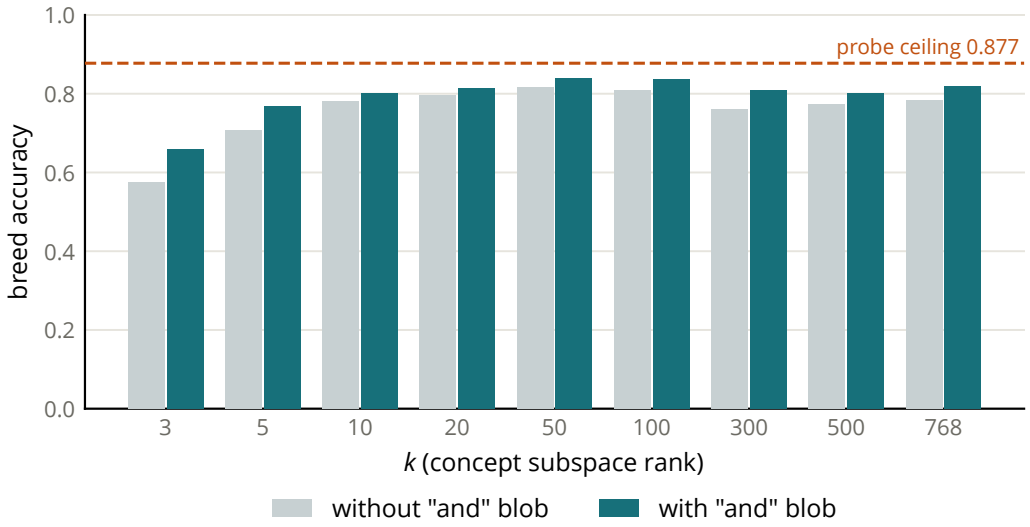

Figure 21: **Real images, same conclusions.** Class accuracy of the recovered components against the probe ceiling (dashed), with and without the "and" blob, over concept subspace rank k.

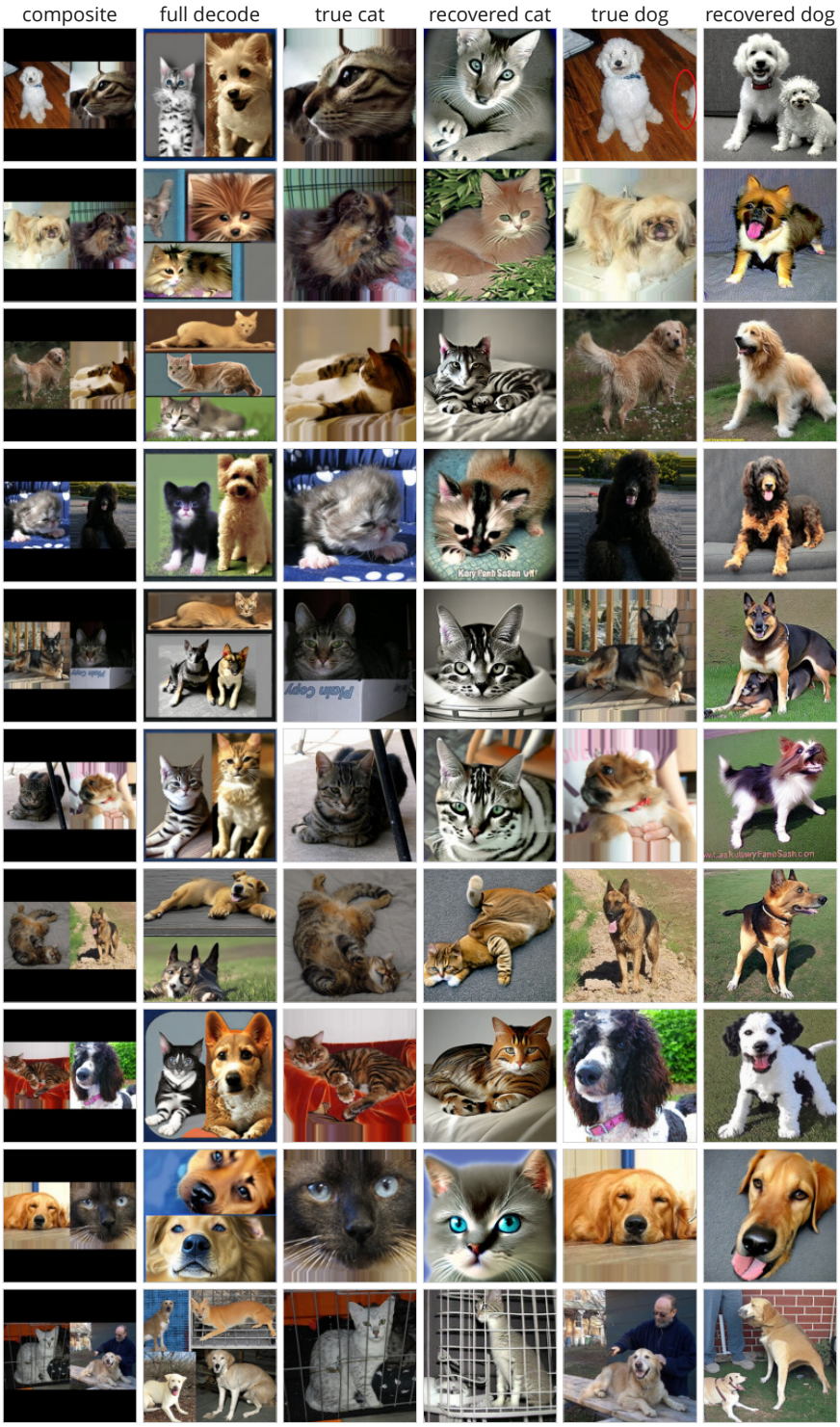

Figure 22: **Decoded components.** Each row shows the composite, an unCLIP decode of its embedding e, and each recovered component $x_c^\star$ beside the true single-animal crop it should match. The components are recovered from the composite embedding alone; no per-object embedding is given to the fit at test time. In the first row, the circled region of the true-dog crop shows a second, partly visible dog, possibly explaining why the recovered dog component (rightmost column) decodes to two dogs rather than one.



Figure 23: **Decomposing generated scenes, 1 of 3.** Hand-selected FLUX.1-schnell scenes, each prompted with a cat and a dog of the breeds named on its row and decomposed with the blobs fitted on ImageNet composites. Each row shows the scene, an unCLIP decode of its full embedding, and decodes of the two recovered components.

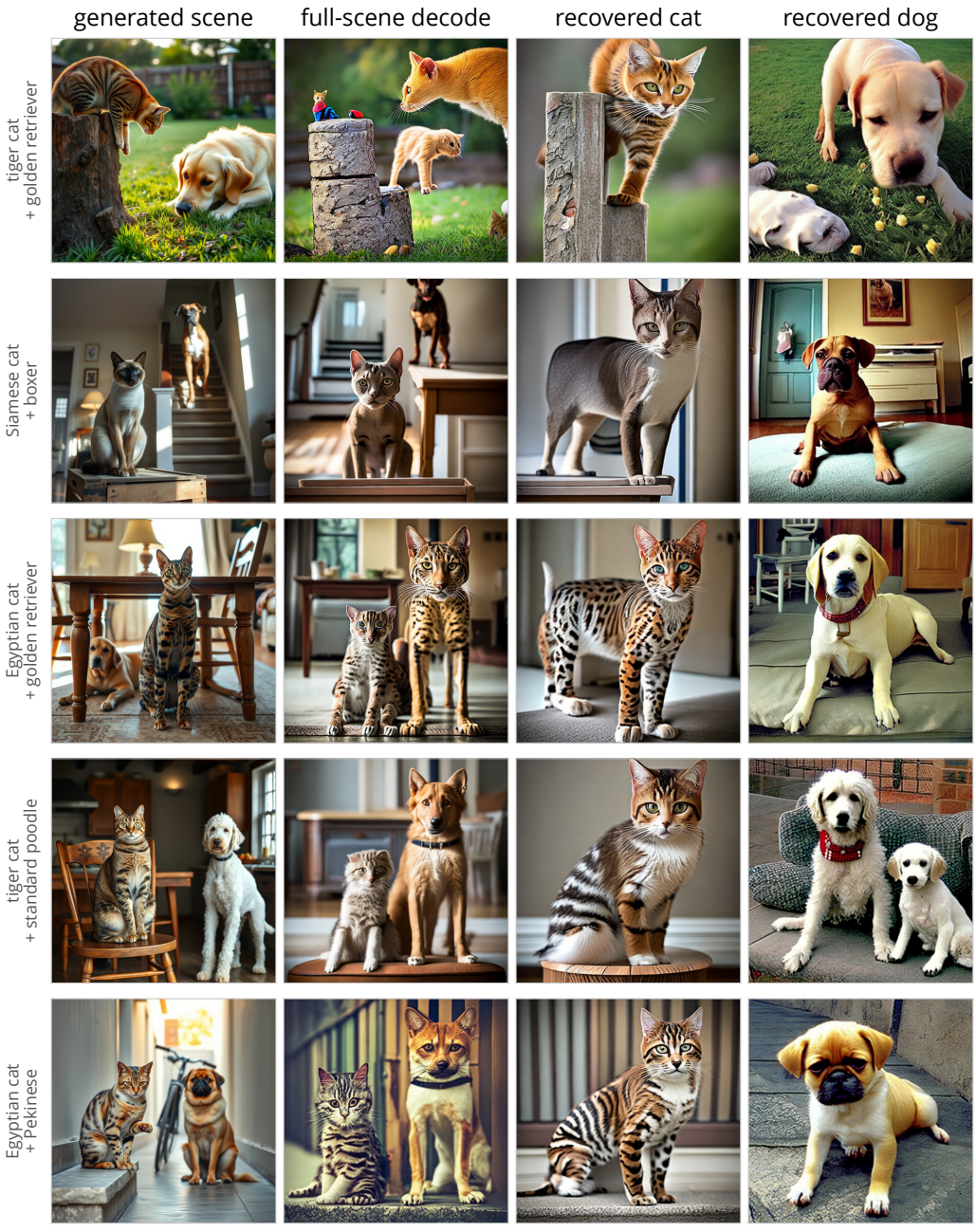

Figure 24: **Decomposing generated scenes, 2 of 3.**



Figure 25: **Decomposing generated scenes, 3 of 3.**