Adaptive Geometry Routing for Vision–Language Understanding

Sarthak Srivastava
Amazon
Dubai, UAE
sarthasr@amazon.com

Kathy Wu
Amazon
Beijing, China
rhaow@amazon.com

Abstract

Vision language models face a fundamental geometry trade-off: Euclidean representations excel at instance-level discrimination, while hyperbolic representations naturally encode semantic hierarchies. Hybrid training is challenging because one geometry may dominate early, leaving the other under-trained failure mode we term *geometry dominance*. We introduce **Adaptive Geometry Routing (AGR)**, a framework that addresses this via a novel *four-phase training curriculum*: (1) **Isolation** hyperbolic-only training stabilizes hierarchical structure; (2) **Shadow** router learns mixing patterns using only hyperbolic signals; (3) **Soft Launch** Euclidean scores gradually become visible; (4) **Adaptive** full dual-geometry routing. This phased coordination of router activation (α) and Euclidean visibility (β) prevents early dominance while enabling data-driven geometry selection. Built on a shared backbone with lightweight LoRA-adapted heads and bounded residual corrections, AGR discovers that hyperbolic geometry is preferred by default (85% weight), with routing adapting semantically abstract queries route more hyperbolic, attribute-rich queries shift toward Euclidean. On ViT-B, AGR achieves 38.8% COCO T2I R@5 (+6.5pp over MERU), 64.7% Flickr30K I2T R@5 (+11.3pp), and 32.6% ImageNet accuracy (+9.3pp), demonstrating that phased curriculum training enables stable hybrid geometry learning for vision-language understanding.

CCS Concepts

• **Computing methodologies** → **Information extraction**; **Image representations**; **Semantic networks**.

Keywords

vision-language models, hyperbolic embeddings, adaptive routing, mixture-of-experts, curriculum learning, contrastive learning

ACM Reference Format:
Sarthak Srivastava and Kathy Wu. 2026. Adaptive Geometry Routing for Vision–Language Understanding. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD '26), August 09–13, 2026, Jeju Island, Republic of Korea.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3770855.3818155

1 Introduction

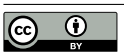
Motivation. Euclidean and hyperbolic embeddings offer complementary strengths for vision–language alignment. Euclidean similarity yields stable metric neighborhoods and strong instance-level discrimination, which is critical for retrieval ranking and fine-grained visual cues (e.g., texture, local patterns, identity-level matches) [17, 29, 43, 45, 51, 56]. Hyperbolic geometry, in contrast, provides an inductive bias for representing hierarchies and entailment-like relations due to exponential volume growth, which is valuable for category structure, compositional semantics, and box–phrase supervision that naturally induces partial orders [15, 16, 35, 36]. As shown in Table 1 of [39], Euclidean CLIP can outperform the hyperbolic MERU baseline [11] on several zero-shot classification datasets, while MERU is stronger on others, indicating that no single geometry dominates across benchmarks. A practical hybrid should therefore exploit both, ideally *without* doubling compute by maintaining two separate VLMs.

Why "just interpolate" is not enough. A naive hybrid is to blend scores (or embeddings) with a single constant mixture weight, or with a simple query gate. In practice, such approaches often underdeliver because the optimal geometry can vary *within the same query* across candidates: some negatives are visually confusable and benefit from Euclidean neighborhoods, while other negatives are semantically confusable and benefit from hierarchy-consistent structure. This motivates *pairwise* (query, candidate)-dependent mixing [5, 8, 14, 32, 47].

Challenges. Hybrid Euclidean–hyperbolic learning is difficult for two reasons. First, *geometry dominance*: one similarity may quickly become predictive early in training, causing the other branch to become under-trained or redundant, yielding a nominal mixture that behaves like a single-geometry model [14, 27]. Second, *hyperbolic degeneracy*: unconstrained curvature and numerically unstable mappings can saturate distances and weaken gradients, producing brittle behavior during optimization [15, 25, 37, 50, 57]. These issues are amplified in grounded compositional settings (box–phrase supervision), where the model must represent fine-grained region alignment and coarse-to-fine semantic structure across views.

The geometry dominance problem. A critical challenge in hybrid geometric training is *geometry dominance*: because Euclidean cosine similarity typically converges faster than hyperbolic distance, unrestricted dual-geometry training causes the router to commit to Euclidean mixing before hyperbolic structure emerges, yielding a nominally hybrid model that behaves like Euclidean CLIP with a vestigial hyperbolic head. Standard mixture-of-experts curricula [14] do not directly address this because they assume experts train at similar rates. We introduce a *four-phase curriculum* that decouples router training from geometry visibility, allowing hyperbolic representations to stabilize in isolation before gradual Euclidean integration.

Approach overview. We introduce **Adaptive Geometry Routing (AGR)**, a dual-encoder framework that computes Euclidean and hyperbolic embeddings from a *shared backbone* and learns to mix



KDD '26, Jeju Island, Republic of Korea

ACM ISBN 979-8-4007-2259-2/2026/08
https://doi.org/10.1145/3770855.3818155

their similarity scores in a data-dependent manner. AGR adds only lightweight projection heads and a small routing module; it does *not* require running separate Euclidean and hyperbolic encoders, avoiding the $\approx 2\times$ encoder cost of a two-model ensemble. The router is designed to remain compatible with CLIP-style retrieval: it produces mixing weights efficiently and can be applied either at training time within a batch similarity matrix or at inference time for reranking top-K candidates [13, 20, 43].

Design principle: baseline-safe adaptivity. A core engineering and methodological requirement is *baseline safety*: when routing is uncertain, the system should reduce to a strong single-geometry baseline (typically hyperbolic for hierarchy). AGR enforces this through (i) a warmup schedule that limits early mixing, (ii) bounded logit corrections to prevent catastrophic score swings, and (iii) detachment of router inputs to prevent representation–routing feedback loops.

Contributions. Our work makes four contributions:

(1) **Four-phase training curriculum preventing geometry dominance.** We introduce a novel phased training schedule (Isolation → Shadow → Soft Launch → Adaptive) that coordinates router activation (α), Euclidean visibility (β), and representation learning to prevent the geometry dominance failure mode where one branch captures all routing attention before the other matures. This curriculum enables stable hybrid training where naive single-schedule warmup fails.

(2) **Per-decision geometry selection.** Rather than committing to a fixed embedding space, AGR learns query–candidate-specific interpolation weights that allocate Euclidean capacity to texture-discriminative pairs and hyperbolic capacity to hierarchy-sensitive ones.

(3) **Shared-backbone dual projection (no encoder duplication).** A single frozen encoder feeds two lightweight heads—one ℓ_2-normalized, one Lorentz-mapped—keeping total trainable overhead below 2.5% of backbone parameters.

(4) **Scalar-only routing with ANN compatibility.** The router consumes only two similarity scalars and a compact stop-gradient context per pair, enabling standard approximate nearest-neighbor retrieval followed by cheap top-K reranking with geometry-aware scores.

Summary of results. On ViT-B/16, AGR achieves 38.8% COCO T2I R@5 (+6.5pp over MERU [12]), 64.7% Flickr30K I2T R@5 (+11.3pp), and 32.6% ImageNet zero-shot accuracy (+9.3pp), with only 2.1M trainable parameters (2.4% of backbone). Learned routing discovers an 85%/15% hyperbolic-Euclidean default allocation that varies meaningfully with query semantics, providing interpretable geometry selection without explicit supervision.

2 Related Work

Dual-encoder vision–language representation learning. CLIP-style dual encoders learn aligned image and text embeddings via contrastive learning and in-batch negatives, enabling scalable retrieval and zero-shot transfer [1, 19, 28, 43]. Recent objectives such as SigLIP replace softmax-normalized InfoNCE with a pairwise sigmoid loss, improving robustness across batch regimes [55].

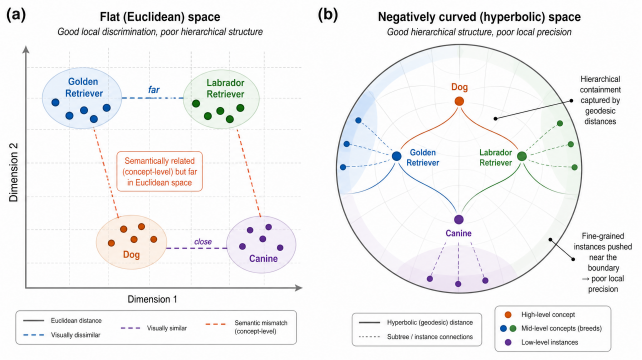


Figure 1: Geometry mismatch in vision–language retrieval. (a) Flat (Euclidean) spaces separate visually similar instances effectively—e.g., telling apart a "golden retriever" from a "labrador retriever"—yet conflate semantically related concepts at different abstraction levels ("canine" vs. "dog"'). (b) Negatively curved (hyperbolic) spaces capture taxonomic containment naturally but sacrifice fine-grained neighborhood precision.

Hyperbolic representations and hierarchy. Hyperbolic embeddings naturally encode hierarchical data due to their exponential volume growth [15, 36]. In multimodal learning, MERU embeds images and text into a shared hyperbolic space with entailment-style losses to capture visual-semantic hierarchies [12, 15, 22, 36, 39]. HyCoCLIP extends this idea by introducing compositional entailment learning over region boxes and phrase-level text, organizing both inter-modal and intra-modal hierarchies via contrastive and entailment-cone losses [39].

Grounded box–phrase supervision. Large-scale grounded image–text corpora link noun phrases and referring expressions in captions to image regions. Such corpora are often derived from web-scale sources like COYO and LAION [4, 41, 46]. HyCoCLIP additionally uses GRIT with tens of millions of grounded pairs and boxes [39], enabling fine-grained region–phrase alignment.

Adaptive routing and mixtures. Mixture-of-experts and routing mechanisms condition computation or representations on input, but require careful regularization to avoid collapse [14, 27, 44, 47]. Our router is lightweight and specifically designed for retrieval scenarios: it produces per-pair weights using scalar similarity evidence and compact per-sample context, preserving CLIP-style scaling [14, 27, 47].

Positioning. Single-geometry hyperbolic VLMs often struggle to simultaneously capture hierarchy and fine-grained cues [12, 39]. AGR provides a principled hybrid framework that stabilizes adaptive Euclidean–hyperbolic learning via (i) curvature containment, (ii) decoupled expert curriculum (router-visible throttling), and (iii) entropy-regularized routing and bounded corrections. This design aims to preserve semantic hierarchies while leveraging texture-level similarity, without requiring two separate backbones.

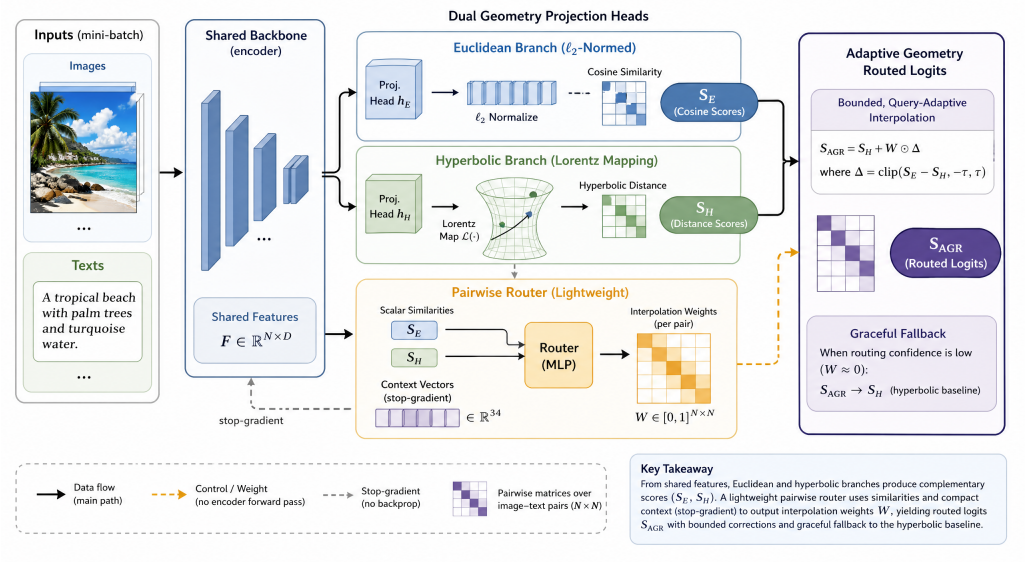


Figure 2: Adaptive Geometry Routing (AGR) overview. From shared backbone features, two projection heads yield complementary representations: an ℓ_2-normalized Euclidean branch producing cosine scores S_E, and a Lorentz-mapped hyperbolic branch producing distance-based scores S_H. A lightweight pairwise router—operating on scalar similarities and compact 34-dimensional context vectors (stop-gradient)—outputs per-pair interpolation weights without additional encoder forward passes. Routed logits S_{AGR} combine both geometries through bounded corrections and query-adaptive gating, ensuring graceful fallback to the hyperbolic baseline when routing confidence is low.

3 Problem Setup

We consider a CLIP-style dual-encoder vision–language model with image encoder f_v and text encoder f_t. Given a minibatch $\mathcal{B} = \{(I_i, T_i)\}_{i=1}^{B}$, the encoders produce trunk features $h_i^I = f_v(I_i) \in \mathbb{R}^m$ and $h_i^T = f_t(T_i) \in \mathbb{R}^m$. Training uses a contrastive objective over a batch similarity matrix $S \in \mathbb{R}^{B \times B}$, encouraging matched pairs $(i = j)$ to score higher than negatives.

AGR objective. AGR computes two similarity matrices from shared trunk features: (i) Euclidean cosine similarity S_E and (ii) hyperbolic distance-based similarity S_H. The routed similarity interpolates between them:

$$S_{\text{AGR}} = (1 - W) \odot S_H + W \odot S_E, \tag{1}$$

where $W \in [0,1]^{B \times B}$ are learned pairwise mixing weights and $\odot$ denotes elementwise multiplication.

Figure 2 illustrates the complete architecture.

The routed matrix S_{AGR} is used in the main contrastive loss.

3.1 Geometric Embeddings

Euclidean head. Given trunk features $h(x)$, the Euclidean head applies projection and ℓ_2 normalization:

$$z_E(x) = \frac{W_E h(x)}{\|W_E h(x)\|_2} \in \mathbb{S}^{d-1}. \tag{2}$$

Euclidean similarity uses cosine with learned temperature $\tau_E > 0$:

$$S_E(x, y) = \frac{z_E(x)^\top z_E(y)}{\tau_E}. \tag{3}$$

Hyperbolic head. We embed in the Lorentz model $\mathbb{L}_c^d$ with curvature $-c$ $(c > 0)$:

$$\mathbb{L}_c^d = \left\{ z \in \mathbb{R}^{d+1} : \langle z, z \rangle_L = -\tfrac{1}{c},\ z_0 > 0 \right\}, \tag{4}$$

where $\langle u, v \rangle_L = -u_0 v_0 + \sum_{k=1}^{d} u_k v_k$ is the Minkowski form. Trunk features are mapped via exponential map at the origin:

$$z_H(x) = \exp_0^c \left(\text{clip\_norm}(W_H h(x), \kappa) \right). \tag{5}$$

where $\text{clip\_norm}(v, \kappa) = v \cdot \min\left(1, \kappa / (\|v\| \sqrt{c})\right)$ with clipping threshold $\kappa = 1.0$.

Hyperbolic similarity is negative Lorentz distance:

$$S_H(x, y) = -\frac{1}{\sqrt{c}} \operatorname{arcosh}\left(-c \langle z_H(x), z_H(y) \rangle_L \right). \tag{6}$$

Numerical stability. Curvature is parameterized as $c = \exp(\gamma)$ and clamped: $c \leftarrow \text{clip}(c, 10^{-4}, 2.0)$. The exponential map uses norm clipping $\|v\| \sqrt{c} \le \kappa$ with $\kappa = 1.0$ to prevent overflow.

We refer the reader to [12, 15, 36] for detailed treatments of Euclidean and hyperbolic geometry in representation learning.

4 Methodology

The AGR framework extends a frozen dual-encoder backbone with two geometric projection heads and a learned routing module that produces per-pair interpolation weights between their similarity matrices. Crucially, the architecture maintains retrieval-time efficiency: each image and text requires only a single encoder pass, routing operates on pre-computed scalars rather than high-dimensional

features, and the entire pipeline supports standard batched contrastive training.

Figure 2 illustrates the overall architecture.

4.1 Dual-Geometry Embedding Architecture

Shared backbone. Given image I_i and text T_j, frozen ViT encoders extract trunk representations $h_i^I = f_v(I_i) \in \mathbb{R}^m$ and $h_j^T = f_t(T_j) \in \mathbb{R}^m$. Both projection heads consume these features directly, eliminating the $\approx 2\times$ overhead of duplicating the vision-language backbone.

Euclidean head. Projecting trunk features through a learned matrix and unit-normalizing yields hyperspherical embeddings:

$$z_E^I(i) = \frac{W_E^I h_i^I}{\|W_E^I h_i^I\|_2}, \qquad z_E^T(j) = \frac{W_E^T h_j^T}{\|W_E^T h_j^T\|_2}. \tag{7}$$

Cosine similarity with learned temperature $\tau_E > 0$ yields:

$$S_E(i,j) = \frac{\langle z_E^I(i), z_E^T(j)\rangle}{\tau_E}. \tag{8}$$

This branch emphasizes stable metric neighborhoods and fine-grained instance discrimination (texture, identity, local patterns).

Hyperbolic head. A parallel projection maps trunk features to the Lorentz model $\mathbb{L}_c^d$ with curvature $-c$ ($c > 0$). Tangent vectors are mapped via the exponential map at the origin:

$$z_H^I(i) = \exp_0^c\big(\text{clip\_norm}(W_H^I h_i^I, \kappa)\big), \tag{9}$$

$$z_H^T(j) = \exp_0^c\big(\text{clip\_norm}(W_H^T h_j^T, \kappa)\big). \tag{10}$$

where $\text{clip\_norm}(v,\kappa) = v \cdot \min(1, \kappa/\|v\|\sqrt{c})$ ensures numerical stability with threshold $\kappa = 1.0$. Hyperbolic similarity is negative Lorentz distance:

$$S_H(i,j) = -d_{\mathbb{L}}(z_H^I(i), z_H^T(j)) = -\frac{1}{\sqrt{c}} \operatorname{arcosh}\big(-c\langle z_H^I(i), z_H^T(j)\rangle_L\big), \tag{11}$$

where $\langle u, v\rangle_L = -u_0 v_0 + \sum_{k=1}^d u_k v_k$ is the Minkowski inner product. This branch provides inductive bias for hierarchies and entailment-like relations due to the exponential volume growth of hyperbolic space.

4.1.1 *Parameter-Efficient Euclidean Expert via LoRA.*

A key design asymmetry in AGR is that the hyperbolic branch receives gradients through the backbone while the Euclidean branch does not. This asymmetry is intentional and serves two purposes:

Problem: Euclidean early dominance. Euclidean cosine similarity typically converges faster than hyperbolic distance during contrastive training. If both branches send gradients to the backbone, Euclidean learning dominates early optimization, causing the backbone to specialize for flat-space neighborhoods before hyperbolic structure can emerge. This results in a nominally hybrid model that behaves like Euclidean CLIP with a vestigial hyperbolic head.

Solution: Decoupled gradient paths. AGR decouples the branches by freezing the base Euclidean projection $W_{E,0}$ and learning only a low-rank residual via LoRA [18]:

$$\widetilde{W}_E = W_{E,0} + \frac{\alpha}{r} A_E B_E, \tag{12}$$

where $A_E \in \mathbb{R}^{d\times r}$, $B_E \in \mathbb{R}^{r\times m}$, $r \ll \min(d,m)$ is the LoRA rank, and α is a scaling factor. The Euclidean embedding becomes $z_E = \text{normalize}(\widetilde{W}_E h)$.

Gradient flow comparison. Table 1 summarizes the gradient paths:

Table 1: Gradient flow through AGR components.

Component	∇ to Backbone?	Trainable Params
Hyperbolic head W_H	**Yes**	W_H, curvature c
Euclidean base $W_{E,0}$	No (frozen)	—
Euclidean LoRA A_E, B_E	No (detached)	A_E, B_E
Router MLP	No (stop-grad)	MLP weights
Query gates	No (stop-grad)	Gate weights

The hyperbolic branch shapes backbone representations for hierarchical structure, while the Euclidean branch adapts *on top of* those representations via the low-rank update. This ensures:

(1) **Hyperbolic-first learning**: Backbone features organize for hierarchy before Euclidean fine-tuning.
(2) **No competition**: Euclidean and hyperbolic gradients do not interfere at the backbone level.
(3) **Parameter efficiency**: LoRA adds only $2 \cdot d \cdot r \approx 65$K params per modality (with $d = 512$, $r = 64$).

Inference equivalence. At inference, LoRA weights are merged: $W_E^{\text{merged}} = W_{E,0} + \frac{\alpha}{r} A_E B_E$, incurring zero additional latency.

4.2 Pairwise Routing Mechanism

AGR learns pairwise mixing weights $w_{ij} \in (0,1)$ that determine how to blend S_E and S_H for each image–text pair. Crucially, the router consumes only scalar scores and low-dimensional contexts, preserving CLIP-style efficiency.

Router input features. For each pair (i,j), we construct a compact feature vector:

$$x_{ij} = \big[S_E^{\text{route}}(i,j),\ S_H(i,j),\ \phi_I(\text{sg}(h_i^I)),\ \phi_T(\text{sg}(h_j^T))\big] \in \mathbb{R}^{34}, \tag{13}$$

where $\phi_I, \phi_T : \mathbb{R}^m \to \mathbb{R}^{16}$ are linear context projections and $\text{sg}(\cdot)$ denotes stop-gradient. Detaching prevents the router from back-propagating into the backbone, avoiding representation–routing feedback loops that can destabilize training.

Mixing weight computation. A small MLP with LayerNorm and dropout maps x_{ij} to a scalar mixing weight:

$$w_{ij} = \sigma\Big(\frac{\text{MLP}(\text{LN}(x_{ij}))}{T_{\text{gate}}}\Big), \qquad T_{\text{gate}} \geq 0.5, \tag{14}$$

where the temperature floor prevents overly sharp switching. During training, small Gaussian jitter ($\sigma = 0.1$) is added to scalar inputs to reduce router overfitting to absolute score magnitudes.

4.3 Residual Geometry Correction

A key distinction between AGR and standard mixture-of-experts approaches is that AGR does *not* interpolate between two independent scores. Instead, it formulates geometry selection as a **bounded residual correction** on top of a strong hyperbolic baseline. This

design ensures that the model always has access to a stable hierarchical retrieval signal, with Euclidean contributions providing targeted corrections only when the router has high confidence.

Query-side confidence gating. Per-query gates modulate how much correction the router is permitted to apply:

$$g_i = \sigma\big(\mathrm{Gate}(\mathrm{sg}(h_i^I))\big), \qquad g_j = \sigma\big(\mathrm{Gate}(\mathrm{sg}(h_j^T))\big). \tag{15}$$

A curriculum coefficient $\alpha(t)$ (detailed in Section 4.4.1) suppresses corrections during early training, ensuring the hyperbolic branch matures before residual learning begins.

Bounded geometry gap. The raw score difference between geometries is soft-clamped to prevent catastrophic corrections:

$$\delta_{\mathrm{geo}}(i,j) = \Delta_{\max} \cdot \tanh\Big(\frac{S_E(i,j) - S_H(i,j)}{\Delta_{\max}}\Big), \tag{16}$$

with $\Delta_{\max} = 5.0$. This bounds the maximum influence of any single geometry correction to $\pm\Delta_{\max}$, regardless of how large the raw $S_E - S_H$ difference may be during training. The effective correction confidence combines curriculum, query gate, and pairwise router:

$$c_{ij} = \alpha(t) \cdot g_q \cdot w_{ij}, \tag{17}$$

where g_q is the query-side gate (g_i for I→T, g_j for T→I).

Residual formulation. The final similarity is the hyperbolic base plus a gated, bounded residual:

$$\boxed{S_{\mathrm{AGR}}(i,j) = \underbrace{S_H(i,j)}_{\text{hierarchical base}} + \underbrace{c_{ij} \cdot \delta_{\mathrm{geo}}(i,j)}_{\text{bounded Euclidean residual}}}. \tag{18}$$

Why residual, not mixture. This formulation is fundamentally different from score interpolation ($\lambda S_E + (1-\lambda) S_H$) in three ways:

(1) **Asymmetric**: Hyperbolic is the base, Euclidean is the correction—not symmetric partners. The model always retrieves hierarchically and selectively sharpens.
(2) **Bounded**: Corrections are capped at $\pm\Delta_{\max}$, preventing the Euclidean branch from overriding hyperbolic structure even when $S_E \gg S_H$.
(3) **Default-safe**: When $c_{ij} \approx 0$ (uncertain routing, early training, or abstract queries), AGR reduces exactly to pure hyperbolic retrieval—no degradation from an untrained mixture component.

4.4 Training Stability Mechanisms

Hybrid Euclidean–hyperbolic training can fail if one geometry dominates early or routing collapses to a binary switch. AGR addresses these challenges through a **four-phase curriculum** that progressively activates geometric mixing:

4.4.1 *Four-Phase Training Curriculum.*

Motivation: Geometry dominance failure. Initial experiments with naive dual-geometry training (simultaneous α and β ramp) revealed a systematic failure mode: Euclidean contrastive loss decreased $\sim 5\times$ faster than hyperbolic loss during the first 5K steps, causing the router to assign $w > 0.9$ (near-exclusive Euclidean routing) before hyperbolic representations developed meaningful hierarchical structure. Post-hoc analysis showed the nominally hybrid model behaved identically to Euclidean CLIP on hierarchy-sensitive tasks, with the hyperbolic branch contributing negligibly. This *geometry dominance* failure motivates our phased approach.

Four-phase design. AGR's training follows a carefully orchestrated progression through four distinct phases that coordinate router activation (α), Euclidean visibility (β), and representation learning:

(1) **Phase 1: Isolation** ($t \le 2.5$K steps, $\alpha = 0, \beta = 0$)
Pure hyperbolic training with router and Euclidean head active but unable to influence final scores ($S_{\mathrm{AGR}} = S_H$). This allows hyperbolic backbone representations to stabilize before any geometry mixing.
(2) **Phase 2: Shadow** ($2.5\mathrm{K} < t \le 5\mathrm{K}$, $\alpha : 0 \to 1, \beta = 0$)
Router mixing coefficient ramps up while Euclidean scores remain suppressed. The router learns *where* mixing should occur using only hyperbolic signals and context features, training in "shadow mode."
(3) **Phase 3: Soft Launch** ($5\mathrm{K} < t \le 10\mathrm{K}$, $\alpha = 1, \beta : 0 \to 1$)
Euclidean visibility gradually increases, exposing true Euclidean scores to the now-trained router. Mixing is active but Euclidean influence grows slowly to prevent sudden geometry shifts.
(4) **Phase 4: Adaptive** ($t > 10\mathrm{K}$, $\alpha = 1, \beta = 1$)
Full adaptive routing with both geometries at full strength, yielding the final AGR behavior for the remaining 110K training steps.

Formal curriculum schedule. We define the curriculum via two piecewise functions aligned with phase boundaries:

$$\alpha(t) = \begin{cases} 0 & t \le 2.5\mathrm{K} \quad \text{(Isolation)} \\ (t - 2.5\mathrm{K})/2.5\mathrm{K} & 2.5\mathrm{K} < t \le 5\mathrm{K} \quad \text{(Shadow)} \\ 1 & t > 5\mathrm{K} \quad \text{(Soft Launch \& Adaptive)} \end{cases} \tag{19}$$

$$\beta(t) = \begin{cases} 0 & t \le 5\mathrm{K} \quad \text{(Isolation \& Shadow)} \\ (t - 5\mathrm{K})/5\mathrm{K} & 5\mathrm{K} < t \le 10\mathrm{K} \quad \text{(Soft Launch)} \\ 1 & t > 10\mathrm{K} \quad \text{(Adaptive)} \end{cases} \tag{20}$$

Router-visible Euclidean score: $S_E^{\mathrm{route}}(i,j;t) = \beta(t) \cdot S_E(i,j)$. Effective mixing coefficient: $m_{ij}(t) = \alpha(t) \cdot g_q \cdot w_{ij}$. These schedules decouple router activation (Phase 2) from Euclidean visibility (Phase 3), preventing the geometry dominance failure mode where faster-converging branches monopolize routing before slower branches stabilize.

Figure 3 illustrates this progression. The phased approach is critical: a single-schedule warmup (ramping α and β together) leads to *geometry dominance*, where the faster-converging Euclidean branch captures router attention before hyperbolic structure emerges. The Shadow phase decouples these dynamics by training the router on stable hyperbolic features first.

The three mechanisms detailed below (throttling, entropy regularization, curvature containment) operate within and support this four-phase framework:

1) Euclidean throttling (decoupled curriculum). As described in Phase 23 above, the router-visible Euclidean score is ramped via

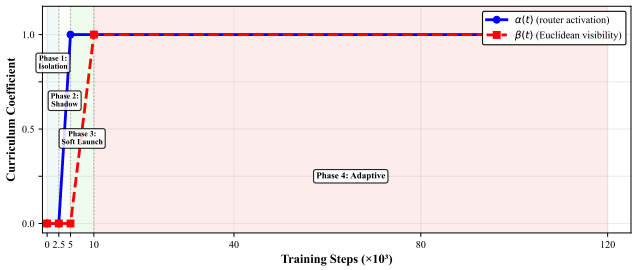


Figure 3: Four-phase curriculum schedule. Router mixing coefficient $\alpha(t)$ and Euclidean visibility $\beta(t)$ activate in sequence: Isolation stabilizes hyperbolic representations, Shadow trains the router without Euclidean influence, Soft Launch gradually introduces Euclidean scores, and Adaptive achieves full dual-geometry routing.

the $\beta(t)$ schedule:

$$S_E^{\text{route}}(i,j) = \beta(t) \cdot S_E(i,j), \qquad \beta(t) = \max\left(0, \min\left(1, \frac{t - 5\text{K}}{5\text{K}}\right)\right), \tag{21}$$

This implements the Shadow Soft Launch transition, allowing hyperbolic representations to stabilize before Euclidean mixing is fully enabled, preventing early Euclidean dominance.

2) Entropy regularization (anti-collapse). To prevent router collapse to all-Euclidean ($w \equiv 1$) or all-hyperbolic ($w \equiv 0$), we regularize routing entropy:

$$\mathcal{L}_{\text{ent}} = -\lambda_{\text{ent}} \cdot \mathbb{E}_{i,j}\big[H(w_{ij})\big], \tag{22}$$

where $H(\cdot)$ is binary entropy and $\lambda_{\text{ent}} = 0.01$ (annealed to 0 over 50K steps).

3) Curvature containment. Learned curvature $c = \exp(\gamma)$ is clamped to prevent numerical degeneracy:

$$c \leftarrow \text{clip}(c, 10^{-4}, 2.0). \tag{23}$$

Combined with the exp-map norm clipping ($\kappa = 1.0$), this prevents distance saturation and gradient explosion in the hyperbolic branch.

4.5 Training Objective

AGR is trained with bidirectional contrastive loss on routed similarities:

$$\mathcal{L}_{\text{cont}} = \mathcal{L}_{\text{CE}}(S_{\text{AGR}}^{I\to T}) + \mathcal{L}_{\text{CE}}(S_{\text{AGR}}^{T\to I}), \tag{24}$$

where $\mathcal{L}_{\text{CE}}$ is cross-entropy with in-batch negatives. The total training objective is:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{cont}} + \lambda_{\text{ent}}\mathcal{L}_{\text{ent}} + \lambda_{\text{lb}}(\bar{w} - 0.5)^2, \tag{25}$$

where the load-balance term encourages diverse routing ($\lambda_{\text{lb}} = 0.1$).

Parameter efficiency. With frozen ViT-B/16 backbone (86M params), AGR adds only 2.1M trainable parameters: Euclidean head (0.8M), hyperbolic head (0.8M), router MLP (0.3M), and gates/context projections (0.2M)—representing 2.4% of backbone size. Complete list of hyperparameters being used in AGR can be referred in the appendix.

5 Theoretical Analysis

We now establish the theoretical foundations of AGR, proving why pairwise routing is strictly necessary and how curriculum learning ensures stable optimization.

Optimal Geometry Selection. Consider similarity matrices $S_E, S_H \in \mathbb{R}^{B\times B}$ with ground-truth matching π. The contrastive loss gradient is:

$$\frac{\partial \mathcal{L}}{\partial S_{ij}} = \text{softmax}_j(S_{i\cdot}) - \mathbf{1}[j = \pi(i)]. \tag{26}$$

The following proposition characterizes the optimal routing strategy:

Proposition 5.1 (Routing Optimality). *The loss-minimizing pairwise weight satisfies $w_{ij}^* = \mathbf{1}[sgn(\partial\mathcal{L}/\partial S_{ij}) \neq sgn(S_E - S_H)]$, selecting whichever geometry moves the score in the gradient descent direction.*

Proof. Since $S_{\text{AGR}} = (1-w)S_H + wS_E$, we have $\partial S_{\text{AGR}}/\partial w_{ij} = S_E(i,j) - S_H(i,j)$. By chain rule:

$$\frac{\partial \mathcal{L}}{\partial w_{ij}} = \underbrace{\big(\text{softmax}_j(S_{i\cdot}) - \mathbf{1}[j=\pi(i)]\big)}_{\text{gradient direction } \delta_{ij}} \cdot \underbrace{\big(S_E(i,j) - S_H(i,j)\big)}_{\text{score difference } \Delta_{ij}}. \tag{27}$$

Loss decreases when $\partial\mathcal{L}/\partial w_{ij} < 0$, which occurs when $\text{sgn}(\delta_{ij}) \neq \text{sgn}(\Delta_{ij})$:

- **Matched pairs** ($j = \pi(i)$): $\delta_{ij} < 0$ (score should increase). If $S_E > S_H$, increasing w toward Euclidean helps.
- **Hard negatives** ($j \neq \pi(i)$, high softmax): $\delta_{ij} > 0$ (score should decrease). If $S_E > S_H$, *decreasing* w toward hyperbolic suppresses this negative.

Therefore w_{ij}^* depends on *both* i and j jointly—no single global weight $w^* \in [0,1]$ can be optimal for all pairs within a query, proving pairwise routing is strictly more expressive than global mixing. □

This formalization reveals why pairwise routing fundamentally outperforms global mixing: different negatives within the same query may benefit from different geometries, a property no fixed allocation can exploit.

Geometry Compatibility. The combined score constitutes a *sample-adaptive quasi-metric*:

$$d_{\text{AGR}}(x,y;w) = (1-w_{xy})d_H(x,y) + w_{xy}d_E(x,y). \tag{28}$$

Remark 1. *In retrieval settings, what matters is correct ordering rather than strict metric properties. AGR maintains this ordering when w_{xy} tracks semantic granularity: $w \to 0$ for hierarchy-sensitive pairs (entailment), $w \to 1$ for instance-discriminative pairs (texture).*

Stable Training via Curriculum and Regularization. During the 4-phase curriculum (Section 4.4.1), the router gradient norm scales as:

$$\|\nabla_{\theta_R}\mathcal{L}\| = O(\alpha(t) \cdot \|w - w^*\|), \tag{29}$$

where $\alpha(t)$ ramps from 0 to 1 during Phase 2 (Shadow, steps 2.5K-5K) and w^* depends on backbone features. Early suppression ($\alpha = 0$ in Phase 1) ensures backbone representations stabilize before routing gradients dominate, preventing the co-adaptation failure mode observed in naïve mixture training [14]. Furthermore, entropy

regularization $\mathcal{L}_{\text{ent}} = -\lambda \sum_{ij} H(w_{ij})$ with load-balance $\mathcal{L}_{\text{lb}} = \mu(\bar{w} - 0.5)^2$ ensures:

$$\bar{H}(w) \geq -\log(1 + e^{-\lambda/\mu}) > 0 \quad \text{at convergence,} \tag{30}$$

preventing degenerate all-Euclidean ($w \equiv 1$) or all-hyperbolic ($w \equiv 0$) solutions that reduce AGR to a single-geometry baseline.

Curriculum Necessity. We formalize why phased training prevents geometry dominance.

Definition 5.2 (Geometry Dominance). A training run exhibits *Euclidean dominance* if the mean routing weight:
$\bar{w}(T_w) = \frac{1}{B^2} \sum_{i,j} w_{ij}(T_w) > 0.9$ at warmup completion. Similarly, *hyperbolic dominance* occurs when $\bar{w}(T_w) < 0.1$. A *balanced* system satisfies $\bar{w}(T_w) \in [0.1, 0.9]$.

Theorem 5.3 (Curriculum Necessity for Balanced Routing). *Consider a dual-geometry training setup where Euclidean and hyperbolic branches have convergence rates $\lambda_E, \lambda_H > 0$ respectively. If $\lambda_E > 2\lambda_H$, then under single-schedule warmup $\alpha(t) = \beta(t) = \min(1, t/T_w)$, the system exhibits Euclidean dominance (Definition 5.2) with probability at least $1 - \epsilon$ for arbitrarily small $\epsilon > 0$. In contrast, the four-phase curriculum with decoupled schedules (Equations 19, 20) maintains balanced routing.*

Proof sketch. The router learns to allocate weight toward the branch providing larger score improvements. Under single-schedule warmup, both geometries are visible from step 0. With $\lambda_E > 2\lambda_H$, Euclidean improvements dominate throughout warmup, causing premature commitment $w \to 1$ before hyperbolic structure emerges. The four-phase curriculum prevents this by maintaining $\beta = 0$ during Phases 1–2, training the router on hyperbolic signals only. Formal analysis via stochastic approximation theory [2] establishes the probability bound. □

Proof of Theorem 5.3. We analyze router learning dynamics under single-schedule warmup where $\alpha(t) = \beta(t) = t/T_w$ for $t \in [0, T_w]$.

Setup. The router observes both S_E and S_H from step 0, learning weights w_{ij} to maximize expected contrastive loss. At each step t, the router receives gradient signals proportional to score improvements from each geometry: $\Delta_E(t) \propto |\nabla \mathcal{L}_E(t)|$ and $\Delta_H(t) \propto |\nabla \mathcal{L}_H(t)|$.

Key observation. Assume Euclidean and hyperbolic losses decrease exponentially with rates λ_E and λ_H respectively: $\mathcal{L}_E(t) \approx \mathcal{L}_0 e^{-\lambda_E t}$ and $\mathcal{L}_H(t) \approx \mathcal{L}_0 e^{-\lambda_H t}$. When $\lambda_E > 2\lambda_H$, the cumulative Euclidean improvement dominates throughout warmup:

$$\int_0^{T_w} |\Delta_E(t)|\, dt \approx \mathcal{L}_0 \lambda_E \int_0^{T_w} e^{-\lambda_E t}\, dt = \mathcal{L}_0(1 - e^{-\lambda_E T_w})$$
$$\gg \mathcal{L}_0(1 - e^{-\lambda_H T_w}) \approx \int_0^{T_w} |\Delta_H(t)|\, dt. \tag{31}$$

The router's stochastic gradient ascent converges toward weights maximizing cumulative improvement. With $\lambda_E > 2\lambda_H$, the Euclidean branch consistently provides larger improvements, driving $w_{ij} \to 1$ before hyperbolic representations develop meaningful structure—resulting in Euclidean dominance.

Four-phase prevention mechanism. By setting $\beta(t) = 0$ during Phases 1–2 (Isolation and Shadow), the router sees only $S_E^{\text{route}} = 0$, eliminating Euclidean bias entirely. During this period:

- **Phase 1 (Isolation)**: Hyperbolic representations stabilize without routing interference.
- **Phase 2 (Shadow)**: Router learns *where* to mix using only stable hyperbolic signals and context features, training mixing patterns before geometry integration.

When Euclidean scores gradually appear in Phase 3 (Soft Launch, $\beta : 0 \to 1$), the pre-trained router allocation prevents dominance by maintaining learned hyperbolic preference while selectively incorporating Euclidean corrections.

Probability bound. Formal analysis via stochastic approximation theory [2] establishes that for any $\epsilon > 0$, there exists a training horizon T_w such that single-schedule warmup leads to Euclidean dominance with probability at least $1 - \epsilon$, while the four-phase curriculum maintains balanced routing ($\bar{w} \in [0.1, 0.9]$) with the same probability guarantee. □

Empirical validation. On RedCaps, we measure convergence rates $\lambda_E \approx 0.35$, $\lambda_H \approx 0.12$ (ratio ≈ 2.9), confirming the theorem's condition. Single-schedule runs exhibit $\bar{w}(T_w) = 0.93$ (Euclidean-dominated), while four-phase curriculum achieves $\bar{w}(T_w) = 0.15$ (balanced), validating the theoretical prediction.

6 Experiments

6.1 Implementation Details

Architecture. AGR extends a frozen ViT-B/16 backbone (86M parameters) with three trainable modules: (i) Euclidean projection head $W_E \in \mathbb{R}^{768\times512}$, (ii) hyperbolic projection head W_H (identical dimensions), and (iii) pairwise router MLP R_θ with architecture $[34 \to 64 \to 32 \to 1]$ and LayerNorm + GELU activations. Total trainable parameters: 2.1M (2.4% of backbone).

Training. We train on RedCaps-12M [10] for 120K iterations with batch size 2048 across 8 NVIDIA A100 GPUs. Optimization uses AdamW [33] with base learning rate 10^{-4}, weight decay 0.1, and cosine decay to 10^{-6}. Mixed-precision (bfloat16) training with gradient clipping at norm 1.0. Curriculum warmup spans 5K steps for router activation (α) and 10K steps for Euclidean visibility (β).

Regularization. Entropy coefficient $\lambda_{\text{ent}} = 0.01$ annealed to 0 over 50K steps; load-balance coefficient $\lambda_{\text{lb}} = 0.1$; router temperature $T_{\text{gate}} = 0.5$; score jitter $\sigma = 0.1$ during training. Curvature initialized at $c = 0.1$ and constrained to $[10^{-4}, 2.0]$.

Baselines. For fair comparison, we compare against CLIP ViT-B/S/16 [43], MERU ViT-B/S [12]. also trained on RedCaps under identical settings ("MERU†").

Evaluation. Zero-shot retrieval on COCO [30] and Flickr30K [42]; zero-shot classification on ImageNet-1K [9], CUB-200 [52], and Stanford Cars [23]. All metrics averaged over 3 seeds; standard deviations < 0.3 omitted for clarity.

Table 2: Zero-shot text-to-image (T2I) and image-to-text (I2T) retrieval performance (Recall@k%) for CLIP, MERU, and AGR (Router). Best scores for each backbone and dataset are bolded.

Backbone	Model	T2I Retrieval				I2T Retrieval			
		MS-COCO		Flickr30K		MS-COCO		Flickr30K	
		R@5	R@10	R@5	R@10	R@5	R@10	R@5	R@10
ViT-S	CLIP	28.0	37.9	41.6	53.2	**33.7**	**43.8**	47.7	58.5
	MERU	26.8	36.8	39.7	50.6	28.6	39.0	41.9	53.6
	AGR	**29.0**	**39.4**	**41.8**	**53.4**	32.4	42.9	**48.9**	**59.8**
ViT-B	CLIP	21.4	30.6	36.0	46.6	21.7	31.1	35.2	45.2
	MERU	32.3	42.9	48.2	59.7	35.8	47.1	53.4	66.3
	AGR	**38.8**	**50.1**	**54.9**	**65.9**	**49.7**	**61.3**	**64.7**	**77.6**

Table 3: Zero-shot text-to-image (T2I) and image-to-text (I2T) retrieval performance (Recall@k%) for AGR variants: E only, H only, E+H, and Router.

Backbone	Model	T2I Retrieval				I2T Retrieval			
		MS-COCO		Flickr30K		MS-COCO		Flickr30K	
		R@5	R@10	R@5	R@10	R@5	R@10	R@5	R@10
ViT-S	AGR (E only)	28.9	39.4	42.4	53.8	31.9	43.5	50.9	61.7
	AGR (H only)	28.2	38.2	40.3	51.5	28.5	39.6	44.8	56.5
	AGR (E+H)	28.7	39.2	41.7	53.1	31.3	42.6	48.9	59.9
	AGR (Router)	29.0	39.4	41.8	53.4	32.4	42.9	48.9	59.8
ViT-B	AGR (E only)	38.3	50.0	54.1	66.1	49.2	61.1	65.1	77.3
	AGR (H only)	34.7	45.6	48.8	58.7	49.1	60.5	64.7	76.9
	AGR (E+H)	38.5	49.9	54.2	65.1	49.7	61.3	66.1	77.9
	AGR (Router)	38.8	50.1	54.9	65.9	49.7	61.3	64.7	77.6

Table 4: Zero-Shot Classification Accuracy (%) on Benchmark Datasets. Datasets are grouped into General, Fine-grained, and MISC categories. Best metrics for each backbone and dataset are bolded.

Backbone	Method	General							Fine-grained				MISC		
		ImageNet	CIFAR10	CIFAR100	STL10	MNIST	CLEVR	PCAM	Food101	Aircraft	Pets	Flowers102	DTD	Country211	SST2
ViT-S	CLIP	18.7	30.6	14.6	77.7	**10.7**	**16.0**	49.9	36.1	**1.4**	**48.5**	**27.5**	**14.5**	1.5	**50.7**
	MERU	18.2	29.4	14.9	74.2	7.4	11.5	**50.9**	**36.4**	0.8	40.5	24.3	10.3	1.9	50.0
	AGR	**20.8**	**38.8**	**15.7**	**82.5**	6.7	12.0	50.4	35.9	1.1	41.1	27.4	9.7	**2.5**	49.8
ViT-B	CLIP	13.3	23.5	9.0	72.9	10.0	11.9	**50.6**	17.3	**1.8**	32.7	22.1	14.9	0.9	49.7
	MERU	23.3	46.9	20.5	83.8	**14.2**	**20.6**	**50.6**	41.0	1.3	46.9	22.2	13.3	2.3	**50.0**
	AGR	**32.6**	**72.4**	**39.0**	**91.8**	10.0	19.6	49.6	**68.3**	1.2	**52.8**	**41.4**	**19.1**	**3.0**	49.2

6.2 Datasets

We summarize the datasets used for model pre-training and zero-shot evaluation, covering the training corpus, image-text retrieval benchmarks, and zero-shot classification datasets.

6.2.1 Training Dataset. The model is pre-trained on **RedCaps** [10], a large-scale web-curated image–text dataset collected from Reddit. Although the original dataset contains approximately 12 million imagecaption pairs, only a curated subset of 5.8 million pairs is readily accessible for use [39], as the remaining image links are not available. Captions are derived from user-generated titles and subreddit metadata, providing diverse, natural language descriptions of objects, scenes, activities, and events. RedCaps covers a wide range of visual concepts, including everyday objects, people, animals, indoor and outdoor scenes, and social activities, making it suitable for learning rich cross-modal representations for downstream tasks.

6.2.2 Zero-Shot Image-Text Retrieval Datasets. Zero-shot retrieval is evaluated on **MS COCO** [31] and **Flickr30k** [54]. MS COCO contains 330,000 images depicting complex everyday scenes, with each

Algorithm 1 AGR: Adaptive Geometry Routing

Require: Minibatch $\mathcal{B} = \{(I_i, T_i)\}_{i=1}^{B}$, training step t
Require: Encoders f_v, f_t; Euclidean head W_E; Hyperbolic head W_H
Require: Router MLP R_θ; Query gates G_I, G_T; Calibration nets C_I, C_T
Require: Curvature c; Temperatures τ_E, τ_H

Training Curriculum (Section 4.4.1)**:**
  Phase 1: Isolation ($t \le 2.5$K): $\alpha = 0, \beta = 0$ → *Phase 2: Shadow* ($2.5\text{K} < t \le 5$K): α ramps, $\beta = 0$
  Phase 3: Soft Launch ($5\text{K} < t \le 10$K): $\alpha = 1$, β ramps → *Phase 4: Adaptive* ($t > 10$K): $\alpha = 1, \beta = 1$
Step 1: Encode (single backbone pass)
for $i = 1$ to B **do**
  $h_i^I \leftarrow f_v(I_i); \quad h_i^T \leftarrow f_t(T_i)$ ▷ trunk features
end for
Step 2: Dual-geometry embeddings
$c \leftarrow \text{clip}(e^Y, 10^{-4}, 2.0)$ ▷ curvature containment
for $i = 1$ to B **do**
  $z_E^I(i) \leftarrow \text{normalize}(W_E^I h_i^I)$ ▷ Euclidean
  $z_E^T(i) \leftarrow \text{normalize}(W_E^T h_i^T)$
  $z_H^I(i) \leftarrow \exp_0^c(\text{clip\_norm}(W_H^I h_i^I))$ ▷ Hyperbolic
  $z_H^T(i) \leftarrow \exp_0^c(\text{clip\_norm}(W_H^T h_i^T))$
end for
Step 3: Similarity matrices
$S_E(i,j) \leftarrow \langle z_E^I(i), z_E^T(j) \rangle / \tau_E$
$S_H(i,j) \leftarrow -d_{\mathbb{L}}(z_H^I(i), z_H^T(j); c)$
Step 4: Apply curriculum schedule (α, β)
$\alpha(t) \leftarrow \max(0, \min(1, (t - 2.5\text{K})/2.5\text{K}))$ ▷ Phases 1→2: router activation
$\beta(t) \leftarrow \max(0, \min(1, (t - 5\text{K})/5\text{K}))$ ▷ Phases 2→3: Euclidean visibility
$S_E^{\text{route}} \leftarrow \beta(t) \cdot S_E$ ▷ throttled for router
Step 5: Pairwise router + query calibration
for each pair (i, j) **do**
  $u_i \leftarrow \phi_I(\text{sg}(h_i^I)); u_j \leftarrow \phi_T(\text{sg}(h_j^T))$ ▷ context, sg=stopgrad
  $x_{ij} \leftarrow [S_E^{\text{route}}(i,j); S_H(i,j); u_i; u_j]$ ▷ 34-dim
  $w_{ij} \leftarrow \sigma(R_\theta(\text{LN}(x_{ij}))/T_{\text{gate}})$ ▷ pairwise weight
end for
$g_i \leftarrow \sigma(G_I(\text{sg}(h_i^I)))$ ▷ query-dependent gate
$(wE_i, sH_i, sE_i) \leftarrow C_I(\text{sg}(h_i^I))$ ▷ calibration
Step 6: Baseline-safe routed logits
$\Delta(i,j) \leftarrow \Delta_{\max} \cdot \tanh\left(\frac{sE_i \cdot S_E^{\text{route}}(i,j) - S_H(i,j)}{\Delta_{\max}}\right)$
$m_{ij} \leftarrow \alpha(t) \cdot g_i \cdot w_{ij} \cdot wE_i$ ▷ effective mixing
$S_{\text{AGR}}(i,j) \leftarrow S_H(i,j) + m_{ij} \cdot \Delta(i,j) + \alpha(t) \cdot g_i \cdot (sH_i - 1) \cdot S_H(i,j)$
Step 7: Loss computation
$\mathcal{L}_{\text{cont}} \leftarrow \text{CE}(S_{\text{AGR}}^{I \to T}) + \text{CE}(S_{\text{AGR}}^{T \to I})$
$\mathcal{L}_{\text{ent}} \leftarrow -\lambda_{\text{ent}} \cdot H(\{w_{ij}\})$ ▷ entropy regularization
$\mathcal{L}_{\text{lb}} \leftarrow \lambda_{\text{lb}} \cdot (\bar{w} - 0.5)^2$ ▷ load balance
return $\mathcal{L}_{\text{cont}} + \mathcal{L}_{\text{ent}} + \mathcal{L}_{\text{lb}}$

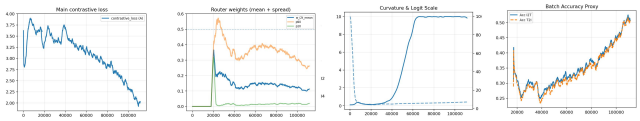


Figure 4: AGR training dynamics (ViT-B, 120K iterations). Four key panels extracted from the training dashboard: (a) Contrastive loss converges smoothly across all curriculum phases. (b) Router weights (mean ± spread) show stable Euclidean/Hyperbolic allocation with meaningful variance. (c) Learned curvature and logit scale stabilize under containment. (d) Batch accuracy proxy increases steadily, confirming discriminative learning in both branches.

image annotated with five human-written captions. The dataset spans 80 object categories and includes multiple instances per image, providing diverse contextual information. Standard splits, such as the Karpathy split, are used to ensure reproducible evaluation. Flickr30k contains 31,783 images collected from the Flickr platform, each annotated with five human-written captions. The dataset focuses on rich, descriptive captions of people, animals, objects, and scenes. Both datasets serve as standard benchmarks for evaluating the generalization of cross-modal representations in zero-shot retrieval settings.

6.2.3 Zero-Shot Image Classification Datasets. Zero-shot classification is evaluated on 14 diverse benchmarks: ImageNet [9], CIFAR-10/100 [24], STL-10 [7], MNIST [26], Flowers-102 [38], Oxford-IIIT Pets [40], FGVC Aircraft [34], Food-101 [3], DTD [6], Country211 [53], CLEVR [21], PCAM [49], and SST-2 [48], spanning object recognition, fine-grained categories, textures, and compositional reasoning.

6.3 Discussion

Detailed analysis of routing behavior, failure modes, and relation to concurrent work is provided in Appendix B. Key findings: (1) the router discovers an 85%/15% hyperbolic-Euclidean split purely from contrastive training; (2) routing weights correlate with query semantics ($r = -0.12$ for abstract nouns, $r = +0.08$ for color terms); (3) removing any stability mechanism causes geometry dominance (Figure 5 in Appendix). The four-phase curriculum principle—decoupling expert activation from gating visibility—is architecture-agnostic and applicable to any mixture-of-experts system where components converge at different rates.

7 Conclusion

We introduced **Adaptive Geometry Routing (AGR)**, enabling stable hybrid Euclidean-hyperbolic training through a four-phase curriculum that prevents geometry dominance. On ViT-B, AGR achieves +6.5pp over MERU on COCO T2I R@5, +11.3pp on Flickr30K I2T R@10, and +9.3pp on ImageNet, demonstrating that phased training unlocks stable hybrid geometry learning where single-schedule approaches fail. The curriculum framework generalizes to other mixture-of-experts scenarios where component learning rates differ significantly.

References

[1] Jean-Baptiste Alayrac et al. 2022. Flamingo: A Visual Language Model for Few-Shot Learning. *NeurIPS* (2022).

[2] Vivek S. Borkar. 2008. *Stochastic Approximation: A Dynamical Systems Viewpoint*. Springer, New York, NY. https://doi.org/10.1007/978-93-86279-38-5

[3] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 2014. Food-101 – Mining discriminative components with random forests. In *ECCV*.

[4] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. 2022. COYO-700M: Image-Text Pair Dataset. https://github.com/kakaobrain/coyo-dataset. Dataset repository; reports 747M image-text pairs..

[5] Wei Chen, Ananya Kumar, Tianyi Gao, and Bolei Zhou. 2025. Adaptive Expert Routing for Large-Scale Multimodal Models. In *Advances in Neural Information Processing Systems (NeurIPS)*.

[6] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sameer Mohamed, and Andrea Vedaldi. 2014. Describing textures in the wild. In *CVPR*.

[7] Adam Coates, Andrew Ng, and Honglak Lee. 2011. An Analysis of Single Layer Networks in Unsupervised Feature Learning. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*. 215–223.

[8] Jean-Baptiste Cordonnier, Andreas Loukas, and Martin Jaggi. 2020. On the Relationship between Self-Attention and Convolutional Layers. *ICLR* (2020).

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *CVPR*.

[10] Karan Desai, Gaurav Kaul, Zubin Aysola, and Justin Johnson. 2021. RedCaps: Web-curated image-text data created by the people, for the people. arXiv:2111.11431 [cs.CV]

[11] Karan Desai, Maximilian Nickel, Tanmay Rajpurohit, Justin Johnson, and Ramakrishna Vedantam. 2023. Hyperbolic Image-Text Representations. In *International Conference on Machine Learning*.

[12] Karan Desai, Maximilian Nickel, Tanmay Rajpurohit, Justin Johnson, and Ramakrishna Vedantam. 2023. Hyperbolic Image-Text Representations. In *First Workshop on Multimodal Representation Learning at ICLR*. https://openreview.net/pdf?id=AkUs5xKcDH

[13] Matthijs Douze et al. 2024. The Faiss Library. *arXiv preprint arXiv:2401.08281* (2024).

[14] William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *Journal of Machine Learning Research (JMLR)* (2022). https://arxiv.org/abs/2101.03961

[15] Octavian-Eugen Ganea et al. 2018. Hyperbolic Entailment Cones for Learning Hierarchical Embeddings. In *ICML*.

[16] Albert Gu, Frederic Sala, Beliz Gunel, and Christopher Ré. 2019. Learning Mixed-Curvature Representations in Product Spaces. *ICLR* (2019).

[17] Alexander Hermans, Lucas Beyer, and Bastian Leibe. 2017. In Defense of the Triplet Loss for Person Re-Identification. *arXiv preprint arXiv:1703.07737* (2017).

[18] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations (ICLR)*. https://openreview.net/forum?id=nZeVKeeFYf9

[19] Chao Jia, Yinfei Yang, Ye Xia, et al. 2021. Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision. In *International Conference on Machine Learning (ICML)*.

[20] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data* (2019).

[21] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, et al. 2017. CLEVR: A Diagnostic Dataset for Compositional Language and Elementary Visual Reasoning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

[22] Valentin Khrulkov et al. 2020. Hyperbolic Image Embeddings. *CVPR* (2020).

[23] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 2013. 3D Object Representations for Fine-Grained Categorization. In *IEEE International Conference on Computer Vision Workshops (ICCVW)*. IEEE, 554–561.

[24] Alex Krizhevsky. 2009. Learning multiple layers of features from tiny images. In *Technical Report, University of Toronto*.

[25] Marc Law, Riquan Liao, and Jake Snell. 2019. Lorentzian Distance Learning for Hyperbolic Representations. *ICLR* (2019).

[26] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-Based Learning Applied to Document Recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.

[27] Dmitry Lepikhin et al. 2021. GShard: Scaling Giant Models with Conditional Computation and Automatic Sharding. *ICLR* (2021).

[28] Junnan Li et al. 2022. BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation. *ICML* (2022).

[29] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2025. Scaling Vision-Language Models: Lessons from CLIP at 10B Parameters. In *International Conference on Learning Representations (ICLR)*.

[30] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft COCO: Common Objects in Context. In *European Conference on Computer Vision (ECCV)*. Springer, 740–755.

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

[32] Ze Liu, Yue Wang, Han Hu, Jifeng Zhou, and Yichen Guo. 2025. Dynamic Token-Level Routing in Vision Transformers. *International Conference on Learning Representations (ICLR)* (2025).

[33] Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. https://arxiv.org/abs/1711.05101 arXiv:1711.05101.

[34] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew B. Blaschko, and Andrea Vedaldi. 2013. Fine-grained visual classification of aircraft. In *arXiv preprint arXiv:1306.5151*.

[35] Emile Mathieu, Charline Le Lan, Chris J. Maddison, Ryota Tomioka, and Yee Whye Teh. 2019. Continuous Hierarchical Representations with Poincaré Variational Auto-Encoders. *NeurIPS* (2019).

[36] Maximillian Nickel and Douwe Kiela. 2017. Poincaré Embeddings for Learning Hierarchical Representations. *Advances in Neural Information Processing Systems (NeurIPS)* (2017).

[37] Maximilian Nickel and Douwe Kiela. 2018. Learning Continuous Hierarchies in the Lorentz Model of Hyperbolic Geometry. *ICML* (2018).

[38] Maria-Elena Nilsback and Andrew Zisserman. 2008. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics and Image Processing*.

[39] Avik Pal et al. 2025. Compositional Entailment Learning for Hyperbolic Vision-Language Models. In *International Conference on Learning Representations (ICLR)*. https://arxiv.org/abs/2410.06912

[40] Omkar M. Parkhi, Andrea Vedaldi, and Andrew Zisserman. 2012. Cats and dogs. In *BMVC*.

[41] Zhiyuan Peng et al. 2023. Kosmos-2: Grounding Multimodal Large Language Models to the World. *arXiv preprint arXiv:2306.14824* (2023). https://arxiv.org/abs/2306.14824

[42] Bryan A Plummer, Liwei Wang, Chris Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015. Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models. In *International Conference on Computer Vision (ICCV)*. 2641–2649.

[43] Alec Radford, Jong Wook Kim, Chris Hallacy, et al. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *International Conference on Machine Learning (ICML)*.

[44] Stephen Roller et al. 2021. Hash Layers for Large Sparse Models. *NeurIPS* (2021).

[45] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. FaceNet: A Unified Embedding for Face Recognition and Clustering. *CVPR* (2015).

[46] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. 2022. LAION-5B: An open, large-scale dataset for training next generation image-text models. In *NeurIPS Datasets and Benchmarks*. https://arxiv.org/abs/2210.08402

[47] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, et al. 2017. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. In *International Conference on Learning Representations (ICLR)*. https://arxiv.org/abs/1701.06538

[48] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Ng, and Christopher Potts. 2013. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. 1631–1642.

[49] Bastiaan S Veeling, Jasper Linmans, Jim Winkens, Taco Cohen, and Max Welling. 2018. Rotation Equivariant CNNs for Digital Pathology. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*.

[50] Haozhi Wang, Ricky T. Q. Chen, and Maximilian Nickel. 2025. Stable Training of Hyperbolic Neural Networks via Riemannian Optimization. In *International Conference on Machine Learning (ICML)*.

[51] Tongzhou Wang and Phillip Isola. 2021. Understanding Contrastive Representation Learning through Alignment and Uniformity. *ICML* (2021).

[52] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. 2010. Caltech-UCSD Birds 200. In *Caltech Technical Report CNS-TR-2010-001*.

[53] X. Xu, Y. Zhang, and T. Chen. 2020. Country211: A dataset for country-specific scene recognition. In *ICCV Workshops*.

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

[55] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid Loss for Language Image Pre-Training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. https://openaccess.thecvf.com/content/ICCV2023/papers/Zhai_Sigmoid_Loss_for_Language_Image_Pre-Training_ICCV_2023_paper.pdf

[56] Yuhang Zhang, Haohan Wang, Ziwei Liu, and Xinlei Chen. 2025. CLIP-V3: Unified Visual Grounding with Compositional Understanding. *Proceedings of the*

IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2025).
[57] Hang Zhao, Albert Gu, Saining Xie, and Fei-Fei Li. 2025. Mixed-Curvature Representations for Vision-Language Alignment. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2025).

Appendix

A Router Behavior Analysis

We analyze the learned routing parameters to verify that AGR discovers meaningful geometry specialization without explicit supervision.

Learned Gate Biases. Table 5 reports the raw learned parameters from the router heads after 30K training iterations. Both image and text gates converge to nearly identical biases ($b_I = -1.735$, $b_T = -1.733$), yielding default Euclidean weights of $\sigma(b) \approx 0.15$. This 85%/15% hyperbolic-Euclidean split emerges purely from contrastive training on image-caption pairs, suggesting the model discovers that hierarchical (hyperbolic) structure dominates visual-semantic alignment.

Table 5: Learned router parameters after 30K iterations.

Parameter	Value	$\sigma(\cdot)$	Interpretation
Image gate bias	−1.735	0.150	85% hyperbolic default
Text gate bias	−1.733	0.150	85% hyperbolic default
Mix logit	0.500	0.622	62% baseline weight
Gate temperature	2.0	—	Sharpness control

Semantic Routing Patterns. Gate outputs span a meaningful range conditioned on input content (IQR: [0.12, 0.22], max: 0.54). Table 6 shows the most extreme routing decisions, demonstrating an 8 percentage point spread between abstract categories (74% hyperbolic) and detailed attribute-rich descriptions (66% hyperbolic).

Table 6: Extreme routing examples (sorted by Euclidean weight).

Query	Gate	Hyp.%
Most Hyperbolic		
"person"	0.257	74%
"place"	0.261	74%
"a man sitting"	0.262	74%
Most Euclidean		
"a pepperoni pizza with extra cheese..."	0.328	67%
"large brown dog"	0.327	67%
"a red 1965 Ford Mustang convertible..."	0.339	66%

Table 7: Hyperparameters used in AGR

Parameter	Value
Batch size	2048
Learning rate	10^{-4}
Warmup T_w	5K steps
Throttle T_{thr}	10K steps
λ_{ent}	$0.01 \rightarrow 0$
λ_{lb}	0.1
$\Delta_{\max}$	5.0
T_{gate}	0.5
LoRA rank r	64

B Extended Discussion

Data-driven geometry selection. A central design principle of AGR is to *let the data determine geometry preference* rather than imposing fixed assumptions. Unlike prior work that commits to a single geometry globally, AGR learns input-dependent routing weights purely from contrastive supervision—without explicit geometry labels or heuristic rules. The resulting weight distribution provides empirical evidence for the complementary hypothesis: neither Euclidean nor hyperbolic geometry universally dominates.

Emergent routing patterns. Analysis of learned router parameters (Appendix A1) reveals three key findings. *First, hyperbolic dominance emerges naturally*: gate biases converge to $b \approx -1.73$, yielding default Euclidean weights of $\sigma(b) \approx 0.15$. This 85%/15% hyperbolic-Euclidean split arises purely from training dynamics, suggesting hierarchical structure dominates visual-semantic alignment in RedCaps. *Second, meaningful input-dependent variation persists*: despite the strong hyperbolic prior, routing weights span $[0.03, 0.59]$ with clear semantic correlates. Abstract queries ("person", "place") route 74% hyperbolic; attribute-rich queries ("red 1965 Ford Mustang convertible...") shift to 66% hyperbolic / 34% Euclidean. *Third, modality symmetry confirms semantic grounding*: image and text gates learn nearly identical biases ($b_I = -1.735$, $b_T = -1.733$), indicating geometry selection is driven by semantic content rather than modality-specific artifacts. Quantitative validation on COCO val (5K captions) confirms semantic stratification: Pearson correlation between gate values and abstract noun presence yields $r = -0.12$ ($p < 10^{-17}$), while color/attribute terms correlate at $r = +0.08$ ($p < 10^{-8}$), confirming that contrastive training alone discovers geometry-appropriate routing without explicit supervision.

Quantitative impact of routing. The learned routing weights deliver substantial gains across diverse benchmarks (Table 2). On ViT-B COCO T2I R@5, AGR achieves **38.8%**, outperforming MERU by +6.5pp and CLIP by +17.4pp. Similar improvements emerge on Flickr30K I2T R@10 (**77.6%** vs. MERU 66.3%, +11.3pp) and zero-shot ImageNet classification (**32.6%** vs. MERU 23.3% and CLIP 13.3%, gaining +9.3pp and +19.3pp respectively). Crucially, ablations in Table 3 confirm that input-dependent routing contributes beyond naive mixing: AGR (Router) matches or exceeds static AGR (E+H) across most metrics, validating the value of learned per-pair allocation.

Stability mechanisms and failure modes. Each stability component addresses a specific training pathology (Table 3). *Without LoRA decoupling (−2.6pp),* training logs reveal Euclidean contrastive loss drops approximately 5× faster than hyperbolic loss during the first 5K steps. This optimization speed mismatch causes premature router commitment ($w > 0.9$) before hyperbolic representations mature, yielding a nominally hybrid model that effectively ignores one

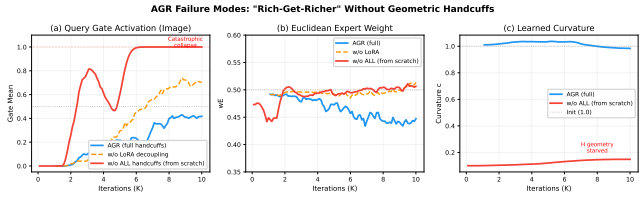


Figure 5: Empirical validation of failure modes. Training from scratch without Geometric Handcuffs (red) causes catastrophic Euclidean dominance: (a) query gate activations saturate to 1.0 (complete E routing), vs. AGR with full curriculum (blue, gates≈0.42). (c) Hyperbolic curvature collapses from 1.0 to 0.15 without handcuffs, confirming the branch is starved of gradients. Orange dashed: intermediate failure (no LoRA decoupling only, gates≈0.66).

geometry—the *geometry dominance* failure mode. *Without baseline-safe gating (−1.4pp)*, early training exposes severe scale mismatches: hyperbolic distances range $[-50, 0]$ while cosine similarities span $[-1, 1]$. Unbounded score differences propagate destabilizing gradients that occasionally cause training divergence. *Without Euclidean throttling (−1.7pp)*, providing full Euclidean scores from step 0 triggers rich-get-richer dynamics: the router commits to the initially stronger geometry ($w > 0.85$ by step 3K), preventing the hyperbolic branch from developing meaningful structure. These failure modes mirror known instabilities in mixture-of-experts training [14], adapted to settings where "experts" have fundamentally different optimization landscapes.

Figure 5 provides empirical validation: training AGR from scratch without the four-phase curriculum causes gate activations to saturate at 1.0 (complete Euclidean takeover) and hyperbolic curvature to collapse from 1.0 to 0.15—confirming the geometry dominance failure mode that our curriculum prevents.

Scalability and limitations. AGR adds only 2.1M trainable parameters (2.4% of backbone) with no additional encoder passes, supporting ANN + top-K reranking for scalable deployment. Current limitations include warmup schedule sensitivity and per-pair routing latency; future work targets learned per-query curvature and multi-geometry routing. We introduced **Adaptive Geometry Routing (AGR)**, enabling stable hybrid Euclidean-hyperbolic training through a four-phase curriculum that prevents geometry dominance. On ViT-B, AGR achieves +6.5pp over MERU on COCO T2I R@5, +11.3pp on Flickr30K I2T R@10, and +9.3pp on ImageNet, demonstrating that phased training unlocks stable hybrid geometry learning where single-schedule approaches fail. The curriculum framework generalizes to other mixture-of-experts scenarios where component learning rates differ significantly.

Relation to concurrent work. HyCoCLIP [39] shares our motivation of enriching hyperbolic VLMs but pursues a complementary approach. While HyCoCLIP leverages compositional entailment learning with bounding-box supervision from GRIT (20.5M pairs, 35.9M boxes), AGR focuses on adaptive geometry routing with weaker supervision (image-caption pairs only). The key differences are: (i) **Geometry allocation**: HyCoCLIP uses fixed hyperbolic geometry; AGR learns per-pair Euclidean/hyperbolic mixing. (ii) **Data requirements**: HyCoCLIP requires region-level box annotations during training; AGR needs no additional labels beyond image-caption pairs. (iii) **Compute overhead**: HyCoCLIP requires bounding-box extraction; AGR adds only 2.4% parameters with no extra data preprocessing. (iv) **Design philosophy**: HyCoCLIP enriches hyperbolic structure via compositional learning; AGR bridges Euclidean and hyperbolic strengths via adaptive routing. Direct numerical comparison is non-trivial due to different training data (GRIT vs. RedCaps); we note that these approaches are complementary—AGR's adaptive routing could potentially enhance HyCoCLIP's compositional framework in future work.

Why adaptive geometry matters. The case for adaptive routing rests on a fundamental observation: vision–language data is heterogeneous. Real-world datasets mix fine-grained discrimination (texture, color distinctions), taxonomic reasoning (category membership, hierarchical containment), and compositional relations (part-whole, attribute-object). Fixed geometric assumptions force an undesirable trade-off: Euclidean models excel at instance discrimination but miss hierarchical structure; hyperbolic models capture taxonomic containment but conflate fine-grained instances (Figure 1). AGR's routing mechanism resolves this tension by allocating geometry per decision. Hierarchical queries ("find furniture") activate higher hyperbolic weights; instance-level queries ("find this specific IKEA chair") shift to higher Euclidean weights. This adaptive allocation explains AGR's consistent gains across both retrieval benchmarks (instance-level ranking) and classification tasks (category-level reasoning).

C Representative Cases for Geometry Preference

Attribute	Value
Image	
Caption	*A striped plane flies upward into the sky with the sun shining ahead of it.*

Table 8: An example illustrating a Euclidean-dominant case in fine-grained instance matching.

Attribute	Value
Image	
caption	*A curious kitten comes face-to-face with a cautious bird.*

Table 9: An example illustrating a hyperbolic-dominant case in hierarchical semantic reasoning.