ALICE: ALIGNING LATENT REASONING WITH CHAIN-OF-THOUGHT FOR MULTIMODAL-LLM EMBEDDINGS

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent multimodal large language model (MLLM) embedders show potential to improve retrieval by generating explicit chain-of-thought (CoT) rationales before the final representation. However, this introduces substantial inference latency and may incur information loss due to discrete autoregressive generation. In this paper, we investigate bringing latent reasoning into multimodal embedding. We first show that latent tokens trained with curriculum-based next-token prediction tend to collapse to predicting a narrow set of CoT-initial tokens, suggesting a bias toward local token prediction over holistic reasoning semantics. Based on this observation, we propose **ALICE** (**A**ligning **L**atent Reason**i**ng with **C**hain-of-Thought **E**mbeddings), a direct *hidden-state to embedding-space* alignment approach: we insert a short block of `<latent>` tokens into the query and train their hidden states to match per-step CoT anchors from a separately trained multi-granularity reference embedder, followed by a free-form contrastive stage that adapts the latents for retrieval. ALICE thus requires a single forward pass at inference, without autoregressive generation. On MMEB, ALICE achieves strong results across the 2B–8B scale, reaching 74.8 with Qwen2-VL-7B, with minimal additional latency. Analysis further reveals that the learned latents exhibit sequential structure, encode instance-specific information, and become increasingly aligned with the final embedding, suggesting reasoning-like behavior in latent space.

1 INTRODUCTION

Multimodal representations underpin a wide range of downstream applications, from multimodal retrieval (Wei et al., 2024; Zhou et al., 2025) to web search (Mitra et al., 2017; Huang et al., 2020) and retrieval-augmented generation (Lewis et al., 2020; Singh et al., 2025). Inspired by the strong comprehension capabilities of multimodal large language models (MLLMs), a growing line of work repurposes MLLMs as embedders (Jiang et al., 2025; 2024; Gu et al., 2025; Lan et al., 2025; Kong et al., 2025; Zhang et al., 2025; Zhu et al., 2026). Unlike CLIP-like dual-tower designs (Radford et al., 2021; Zhai et al., 2023; Sun et al., 2023; Li et al., 2022), an MLLM embedder ingests arbitrary interleaved multimodal inputs and maps them to a shared space via the hidden state of a designated output token, yielding consistent gains across diverse benchmarks. Recent methods further show that *reasoning before embedding* can substantially improve representation quality, either by routing the query through an external chain-of-thought (CoT) (Wei et al., 2022) reasoner (Cui et al., 2026; Jiang et al., 2026) or by letting the embedder itself generate a reasoning path before the final readout (Lan et al., 2026; Liu et al., 2025b; Hao et al., 2026).

Although effective, embedding with explicit CoT introduces two fundamental drawbacks. First, it *incurs significant overhead*: either routing through an external reasoner (Cui et al., 2026; Jiang et al., 2026) or generating lengthy rationales ranging from roughly 60 tokens (Liu et al., 2025b) to over 400 tokens (Lan et al., 2026), inflating per-query cost. Second, it can be *noisy*: discrete sampling may lose information or introduce hallucinations that affect the final embedding (Liu et al., 2025b).

A natural idea is to retain the benefits of reasoning while avoiding the discrete-sampling overhead. In the language-only regime, continuous or *latent* reasoning has offered a promising alternative. Methods such as COCONUT (Hao et al., 2025), CODI (Shen et al., 2025), and SIM-CoT (Wei et al., 2026) replace discrete CoT tokens with continuous "thought" vectors in the hidden space, often matching explicit CoT while dramatically shortening the reasoning chain. Porting this idea to multimodal

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

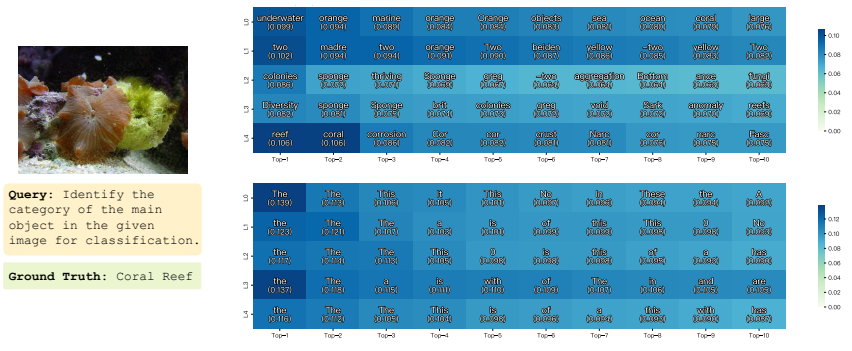


Figure 1: **Decoding the <latent> hidden states** (5 tokens, for illustration). Top: ALICE (ours) after our proposed CoT alignment. Bottom: a COCONUT-like model. The COCONUT-like latents decode to the first tokens of subsequent CoT steps, while ALICE's vocabulary projections reveal richer semantic content beyond local next-token predictions.

embedding, however, is far from trivial. These methods are designed for autoregressive generation: the latent hidden states are implicitly shaped by the *next discrete token prediction* objective, as downstream tokens attend to them for contextual information, and the continuous vectors merely serve as a warm-up before the model switches back to language. For embedding tasks, where the goal is a single continuous representation rather than text generation, next-token supervision emphasizes local token predictability and only indirectly guides retrieval-oriented reasoning representations.

To investigate this mismatch, we first construct a stepwise multimodal reasoning corpus by prompting a frontier MLLM to produce multi-step CoT rationales for each query–target pair (Sec. 3.2), then reimplement the COCONUT-style curriculum in our embedding setting, progressively replacing each CoT step with latent tokens while using next-token and contrastive losses jointly (Sec. 4.1). This pilot study reveals a characteristic failure mode. As shown in Fig. 1, when we decode the hidden states of the latent tokens, they consistently predict the *first tokens of subsequent CoT steps*— suggesting that the training objective prioritizes step-initial prediction at the expense of holistic reasoning semantics. To quantify this, we count the first token of each CoT step across our constructed training data and find 3,441 unique surface forms, with the top-500 tokens covering 99.1% of all occurrences. When we try to decode each of the 5 latent tokens to obtain its top-5 predicted tokens, the average overlap with this set of CoT step-initial tokens reaches 79.8% (Fig. 2), showing a strong concentration of their vocabulary predictions on CoT step-initial tokens. We observe a similar concentration in PLUME (He et al., 2026): 58.4% of its latent states' top-5 vocabulary predictions fall within a set comprising the 500 most frequent CoT step-initial tokens and three ending tokens (88.3% for top-3 predictions). Details are provided in Appendix A. This observation suggests that autoregressive curricula, which still tie the latent representations to discrete token prediction, risk emphasizing a narrow set of next-token predictions over holistic reasoning semantics. Since embedding tasks require no text decoding and only demand a single continuous representation, a natural question arises: can we bypass autoregressive supervision and instead align the latent tokens *directly* with the semantics that CoT reasoning is meant to capture?

To this end, we propose **ALICE** (**A**ligning **L**atent Reason**i**ng with **C**hain-of-Thought **E**mbeddings), which *directly* aligns each latent token's hidden state with the semantic embedding space, rather than letting it be implicitly shaped by next-token prediction. We insert a fixed-length block of `<latent>` tokens after every query and align their hidden states with per-step semantic anchors from a separately trained multi-granularity reference embedder. A subsequent free-form contrastive stage further adapts the latents for task-specific retrieval. Once trained, the model requires no autoregressive sampling at inference: ALICE supports latent reasoning for multimodal embedding in a single forward pass, with only a small constant overhead over a vanilla MLLM embedder.

We summarize our contributions as follows.

1. We identify that curriculum-based next-token supervision for latent reasoning risks collapsing to predicting a narrow set of CoT step-initial tokens in the embedding regime, and

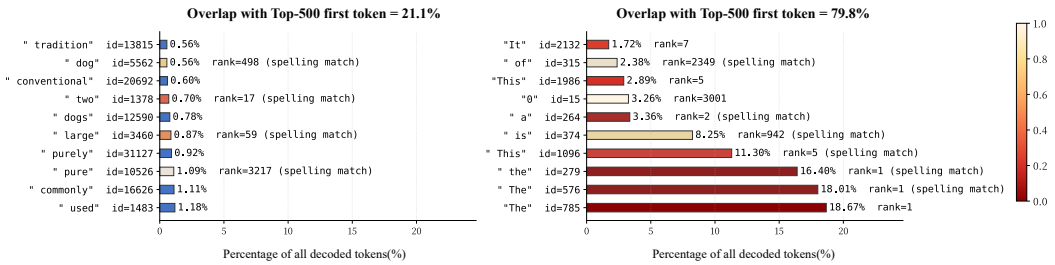


Figure 2: **First-token coverage analysis.** Left: ALICE (ours). Right: a COCONUT-like baseline. For every `<latent>`, we decode their top-5 predicted tokens and measure overlap with CoT step-initial tokens (counted on ImageNet-1K evaluation subset). Red gradient indicates log-rank in the CoT first-token frequency distribution (darker = higher frequency); blue denotes content words absent from the first-token set. The high coverage reveals a strong bias toward next-step-start prediction under autoregressive latent supervision.

therefore propose to explicitly align `<latent>` token hidden states with the semantic space of each CoT step, providing a training signal better suited to multimodal embedding.

2. We develop a two-phase training pipeline—CoT-anchored latent alignment followed by free-form contrastive adaptation—supported by a multi-granularity reference embedder that provides step-level semantic anchors, providing a semantic-alignment approach to latent reasoning for multimodal embedding *without* any autoregressive rollout at inference time.

3. On the MMEB benchmark (Jiang et al., 2025), ALICE achieves strong performance across the 2B–8B parameter scale with minimal additional latency.

2 Related Work

MLLM-based multimodal embeddings. Traditional multimodal representation learning is dominated by dual-tower contrastive models such as CLIP (Radford et al., 2021), SigLIP (Zhai et al., 2023), and their follow-ups (Sun et al., 2023; Li et al., 2022), trained with a symmetric InfoNCE (He et al., 2020) loss. While effective for paired image–text retrieval, these designs struggle with instruction-following queries, composed inputs, and fine-grained task conditioning (Schrodi et al., 2025). In contrast, MLLMs naturally handle interleaved multimodal inputs and complex instructions, motivating recent work that repurposes them as *unified* embedders (Jiang et al., 2024; 2025; Gu et al., 2025; Zhang et al., 2025; Lan et al., 2025; Kong et al., 2025; Zhou et al., 2025; Chen et al., 2025; Liu et al., 2025a; Zhu et al., 2026), extracting the hidden state of a designated output token as the final embedding. ALICE inherits this paradigm but equips the embedder with a *continuous* reasoning prefix that operates entirely in hidden space.

Reasoning large language models. Chain-of-thought (CoT) prompting (Wei et al., 2022) and reinforcement-learning-based reasoning (Guo et al., 2025; Huang et al., 2025) have shown that *explicit* intermediate reasoning substantially improves performance on various tasks (Ke et al., 2025). However, explicit CoT requires long token rollouts and forces the model to externalize every intermediate step as a discrete token. To alleviate this, a recent line of work advocates continuous reasoning in latent space. Earlier work on Pause Tokens (Goyal et al., 2024) appends learnable tokens to the input, allowing additional computation before committing to an answer. COCONUT (Hao et al., 2025) replaces discrete CoT tokens with continuous thought vectors by feeding hidden states back as input embeddings, using a curriculum that progressively substitutes language tokens with latent ones. Follow-ups such as CODI (Shen et al., 2025) and SIM-CoT (Wei et al., 2026) further introduce step-level supervision for the latent hidden states, suggesting that stronger guidance can be beneficial. These methods, however, share a common training paradigm: the latent hidden states are implicitly shaped by the *next discrete token prediction* objective, as downstream tokens attend to them for contextual information. Such a signal suits downstream generation but may not be well-suited for embedding tasks, where the goal is a single continuous representation rather than text generation.

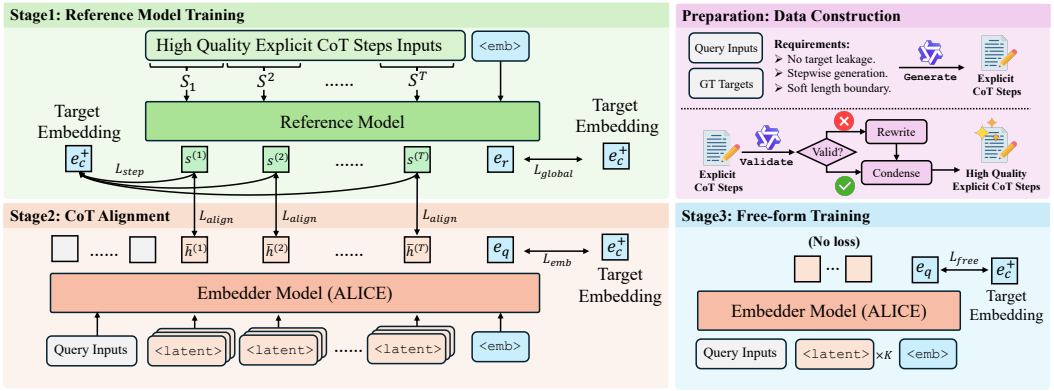


Figure 3: **Overview of ALICE.** We construct stepwise CoT rationales for each query–target pair and train a reference model to project each step into the embedding space. During alignment, the latent tokens in f_θ are grouped and their pooled hidden states aligned with per-step CoT embeddings. The free-form stage removes alignment supervision and trains the model with only a contrastive objective.

Reasoning for MLLM-based embeddings. A growing body of work combines reasoning with embedding. TTE (Cui et al., 2026) and Embed-RL (Jiang et al., 2026) route the query through an external reasoner whose rationale is concatenated to the input before the final readout. UME-R1 (Lan et al., 2026), RGE (Liu et al., 2025b), and TRACE (Hao et al., 2026) instead let the embedder *itself* autoregressively generate a rationale and then use the last-token hidden state as the embedding. While these methods consistently improve over non-reasoning baselines, they inherit the cost of discrete CoT—either an external large-scale reasoner or internal autoregressive generation at inference time. Douyin Multimodal Embedding (Chen et al., 2026) supplements next-token prediction with multi-token prediction in reconstruction to capture longer-range semantics, echoing our motivation for richer semantic supervision of latent reasoning. PLUME (He et al., 2026) combines autoregressive latent rollouts with a semantic-anchor-guided mixture-of-experts transition adapter and a progressive explicit-to-latent curriculum. ALICE instead directly aligns latent representations with stepwise CoT embeddings and forms the retrieval representation in a single forward pass.

3 Method

Our goal is to equip an MLLM embedder with continuous, step-aware latent reasoning without any autoregressive rollout at inference time. To this end, ALICE transfers stepwise CoT semantics into latent representations through direct alignment in the retrieval embedding space. The overall pipeline is illustrated in Fig. 3 and consists of three components: (i) construction of a stepwise multimodal reasoning corpus (Sec. 3.2); (ii) training of a multi-granularity reference embedder that provides per-step semantic anchors (Sec. 3.3); and (iii) a two-phase training procedure that first aligns the embedder's latent tokens to those anchors and then adapts them with a free-form contrastive objective (Sec. 3.4 and 3.5).

3.1 Problem Setup and Notation

Let $\mathcal{D} = \{(q_i, c_i^+)\}_{i=1}^N$ denote a multimodal retrieval corpus, where every query q_i and target candidate c_i^+ can be an arbitrary interleaved combination of text and images. We append a special <emb> token to every input and take its last-layer hidden state as the output embedding. We denote by $f_\theta(\cdot)$ the MLLM embedder with parameters θ, and by $e_\theta(\cdot) = \mathrm{norm}(f_\theta(\cdot)[\texttt{<emb>}])$ the resulting ℓ_2-normalized embedding. Given a batch of size B, the standard in-batch InfoNCE loss is

$$\mathcal{L}_{\text{InfoNCE}}(\{\mathbf{a}_i\}, \{\mathbf{b}_i\}) = -\frac{1}{B}\sum_{i=1}^{B} \log \frac{\exp(\mathbf{a}_i^\top \mathbf{b}_i/\tau)}{\sum_{j=1}^{B} \exp(\mathbf{a}_i^\top \mathbf{b}_j/\tau)}, \tag{1}$$

where τ is the temperature, $\mathbf{b}_i$ is the positive for $\mathbf{a}_i$, and $\{\mathbf{b}_j\}_{j\neq i}$ are in-batch negatives.

To enable continuous reasoning, we augment the query input with K special <latent> tokens inserted right before the <emb> slot:

$$q_i \rightarrow [\, q_i;\ \underbrace{\texttt{<latent>} \cdots \texttt{<latent>}}_{K \text{ tokens}};\ \texttt{<emb>}\,].$$

The hidden states at the K latent positions, denoted $\mathbf{h}_i \in \mathbb{R}^{K\times d}$, are produced in a single forward pass together with the embedding $e_\theta(q_i)$. Unlike Hao et al. (2025); Shen et al. (2025); Wei et al. (2026), where $\mathbf{h}_i$ is implicitly shaped by the *next-token prediction* objective through downstream token attention, we train $\mathbf{h}_i$ to live in the *same* space as the target embeddings.

3.2 Stepwise Reasoning Data Construction

We build our training data on top of the MMEB (Jiang et al., 2025) training split. For each (q_i, c_i^+), we prompt a frontier MLLM (`Qwen3VL-235B-A22B-Instruct`) to produce a multi-step chain-of-thought (CoT) rationale $r_i = (s_i^{(1)}, s_i^{(2)}, \ldots, s_i^{(T_i)})$ that bridges q_i to c_i^+, with each step capturing a complementary reasoning aspect. To ensure correctness, we provide the ground-truth target to the MLLM as a reference during generation. However, the prompt strictly forbids the model from directly referencing the target or reasoning backward from the answer; only information derivable from q_i may appear in r_i. We also impose a soft word budget on each step.

After generation, we apply rule-based filtering to discard any rationale with more than ten steps or any single step longer than 300 tokens, yielding 211 rejected samples. We then ask the MLLM to review each rationale according to three rules: (i) decide whether (q_i, r_i) can jointly lead to c_i^+; if yes, mark r_i as *valid*; (ii) if r_i is valid and $T_i > 5$, check for redundancy and condense redundant rationales to at most five steps while retaining non-redundant ones; (iii) if r_i is invalid, regenerate a new r_i within five steps under the same no-leakage constraint. After this pass, 237 samples are dropped and 81,600 are rewritten. The refinement prompt requests at most five steps for rewritten rationales. Before training, we normalize the annotations to five supervision steps using the sampling and padding procedure in Sec. 4.1.

3.3 Multi-Granularity Reference Embedder

As shown in Fig. 2, without a reliable semantic anchor the embedder's latent hidden states risk collapsing to predicting a narrow set of CoT step-initial tokens. We therefore train a reference embedder f_ϕ that, for every sample, produces a global embedding $e_\phi(r_i)$ over the full CoT and five per-step embeddings $e_\phi(s_i^{(k)})$, all living in the same space as the target $e_\phi(c_i^+)$. The reference model shares the same backbone as f_θ and encodes the full textual CoT r_i, each individual step $s_i^{(k)}$, and the multimodal target c_i^+, appending <emb> to each input for readout.

The reference model is optimized with two InfoNCE losses at different granularities:

$$\mathcal{L}_{\text{global}} = \mathcal{L}_{\text{InfoNCE}}\big(\{e_\phi(r_i)\}, \{e_\phi(c_i^+)\}\big)\,, \tag{2}$$

$$\mathcal{L}_{\text{step}} = \mathcal{L}_{\text{InfoNCE}}\Big(\{e_\phi(s_i^{(k)})\}_{i,k\in\mathcal{V}}, \{e_\phi(c_i^+)\}\Big)\,, \tag{3}$$

$$\mathcal{L}_{\text{ref}} = \mathcal{L}_{\text{global}} + \lambda_{\text{step}}\,\mathcal{L}_{\text{step}}, \tag{4}$$

where $\mathcal{V}$ indexes the steps in the batch and the positive of $s_i^{(k)}$ is the target c_i^+ of the same sample. We use $\lambda_{\text{step}} = 0.5$. $\mathcal{L}_{\text{global}}$ aligns the full-rationale semantics with the target, while $\mathcal{L}_{\text{step}}$ ensures each individual step already points toward c_i^+, making every $e_\phi(s_i^{(k)})$ a usable anchor rather than merely a fragment of the global embedding. Since both objectives share the same target space, the per-step embeddings can directly supervise the embedder's latents without extra projection.

3.4 CoT-Anchored Latent Alignment

With a trained reference model in hand, we inject K <latent> tokens after every query of the embedder f_θ and train them to match the reference model's per-step CoT anchors. For sample i, we split $\mathbf{h}_i \in \mathbb{R}^{K\times d}$ into T contiguous chunks of $\lfloor K/T \rfloor$ tokens each and mean-pool each chunk:

$$\bar{\mathbf{h}}_i^{(k)} \;=\; \frac{1}{|\mathcal{I}_k|}\sum_{t\in\mathcal{I}_k} \mathbf{h}_i[t,:], \qquad k = 1, \ldots, T, \tag{5}$$

where $\mathcal{I}_k$ is the index set of chunk k. All evaluated settings use K divisible by T. We then ℓ_2-normalize each $\bar{\mathbf{h}}_i^{(k)}$ and align it with the corresponding reference-model anchor $e_\phi(s_i^{(k)})$ via an InfoNCE loss across the whole batch:

$$\mathcal{L}_{\text{align}} = \mathcal{L}_{\text{InfoNCE}}\Big(\{\bar{\mathbf{h}}_i^{(k)}\}_{i,k}, \{e_\phi(s_i^{(k)})\}_{i,k}\Big), \tag{6}$$

The reference model is frozen during this stage. To prevent the <emb> hidden state from drifting, we additionally impose a contrastive loss on the final query embedding:

$$\mathcal{L}_{\text{emb}} = \mathcal{L}_{\text{InfoNCE}}\big(\{e_\theta(q_i)\}, \{e_\theta(c_i^+)\}\big), \tag{7}$$

and the overall training loss of this stage is

$$\mathcal{L}_{\text{align-total}} = \mathcal{L}_{\text{align}} + \mathcal{L}_{\text{emb}}. \tag{8}$$

During this stage, stepwise CoT supervision is applied to the K <latent> tokens, which matches the inference-time input format. We use $K = 15$ latent tokens (i.e., three per reasoning step) throughout this work; see Sec. 4.3 for an ablation on the number of latent tokens.

3.5 Free-Form Contrastive Training

The alignment stage initializes each latent group with supervision from its corresponding CoT step. Its representations remain constrained by the reference anchors. We then release the latents from the alignment supervision and let them reshape freely for retrieval. Concretely, we remove $\mathcal{L}_{\text{align}}$ and train f_θ end-to-end with the single contrastive objective:

$$\mathcal{L}_{\text{free}} = \mathcal{L}_{\text{InfoNCE}}\big(\{e_\theta(q_i)\}, \{e_\theta(c_i^+)\}\big). \tag{9}$$

The latents retain their step-aware initialization while adapting to the retrieval objective; the <emb> token reads them out in a single forward pass. At inference, the model appends K <latent> tokens and one <emb> token to the query, extracting the embedding in a single forward pass.

4 Experiments

4.1 Setup

Evaluation protocol. We evaluate on MMEB (Jiang et al., 2025), a multimodal retrieval benchmark comprising 36 datasets across four task types: classification, VQA, retrieval, and visual grounding. Each dataset is formulated as a similarity-based ranking task; we report Hit@1 following the standard protocol. MMEB spans 20 in-domain (IND) and 16 out-of-domain (OOD) tasks, covering both in-distribution and generalization evaluation.

Implementation details. We adopt the Qwen series (Qwen2-VL-2B/7B, Qwen2.5VL-3B/7B, Qwen3VL-2B/4B/8B) as the backbone. The reference model shares the same backbone and is trained on (r_i, c_i^+) pairs—text-only CoT input paired with multimodal target. Following VLM2Vec (Jiang et al., 2025), we train the reference embedder, the latent-alignment stage, and the free-form stage for one epoch each on the MMEB training split with a global batch size of 1,024, a learning rate of 2×10^{-5}, and $\tau = 0.03$ for all contrastive losses. For both ALICE and the COCONUT-like baseline in our ablation, we fix $T = 5$ reasoning steps. During training, rationales shorter than T steps are padded by repeating the last step, while longer rationales are subsampled to T steps, always retaining the final step (typically the conclusion).

For the COCONUT-like baseline, we adopt a curriculum training strategy that progressively replaces CoT steps with latent tokens over T=5 epochs, substituting one CoT step per epoch. The training loss combines a next-token prediction loss (cross-entropy) on the remaining discrete CoT steps with a contrastive loss on the final <emb> token at equal weight; in the last epoch, where all steps have been replaced, only the contrastive loss remains. All other training settings follow ALICE.

4.2 Main Results

ALICE achieves strong retrieval performance with single-pass latent reasoning (Table 1). On MMEB, it reaches 72.8 with Qwen3-VL-4B and 74.8 with Qwen2-VL-7B, exceeding Embed-RL (Jiang et al., 2026) and UME-R1 (Lan et al., 2026) on the same respective backbones by 1.6 and 3.6 points. These gains are achieved without an external reasoner or autoregressive generation at inference, with minimal additional latency over a standard embedder (Table 2).

Table 1: **Performance comparison on the MMEB (Jiang et al., 2025) benchmark.** We report Hit@1, with the best results in **bold** and second-best underlined in each column.

Model	MLLM Backbone	Per Meta-Task Score				Average Score		
		Classification	VQA	Retrieval	Grounding	IND	OOD	Overall
Baseline Models								
CLIP (Radford et al., 2021)	-	42.8	9.1	53.0	51.8	37.1	38.7	37.8
SigLIP (Zhai et al., 2023)	-	40.3	8.4	44.1	59.5	32.3	38.0	34.8
UniIR (Wei et al., 2024)	-	42.1	15.0	60.1	62.2	44.7	40.4	42.8
Magiclens (Zhang et al., 2024)	-	38.8	8.3	35.4	26.0	31.0	23.7	27.8
2B~4B Model Size								
VLM2Vec (Jiang et al., 2025)	Qwen2-VL-2B	58.6	49.2	65.0	73.1	64.8	53.4	59.7
VLM2Vec (Jiang et al., 2025)	Phi-3.5-V-4.2B	54.8	54.9	62.3	79.5	66.5	52.0	60.1
UNITE (Kong et al., 2025)	Qwen2-VL-2B	63.2	55.9	65.4	75.6	65.8	60.1	63.3
UniME(Gu et al., 2025)	Phi-3.5-V-4.2B	54.8	55.9	64.5	81.8	68.2	52.7	64.2
LLaVE (Lan et al., 2025)	Aquila-VL-2B	62.1	60.2	65.2	84.9	69.4	59.8	65.2
UME-R1 (Lan et al., 2026)	Qwen2VL-2B	64.8	62.8	67.6	77.2	71.5	60.4	66.6
Embed-RL (Jiang et al., 2026)	Qwen3VL-2B	62.8	67.9	68.6	90.4	69.5	**68.9**	69.2
RGE (Liu et al., 2025b)	Qwen2.5VL-3B	64.4	67.8	70.2	90.1	75.3	63.7	70.1
Embed-RL (Jiang et al., 2026)	Qwen3VL-4B	63.7	**70.5**	71.3	**91.4**	74.3	67.3	71.2
PLUME (He et al., 2026)	Qwen2-VL-2B	66.5	59.2	67.6	79.7	69.5	62.1	66.3
ALICE (ours)	Qwen2-VL-2B	69.2	65.3	71.9	88.1	75.3	65.9	71.1
ALICE (ours)	Qwen3VL-2B	68.7	64.7	70.5	90.6	75.2	64.9	70.6
ALICE (ours)	Qwen2.5VL-3B	66.8	68.7	72.4	90.6	77.0	65.4	71.9
ALICE (ours)	Qwen3VL-4B	**69.5**	68.6	**73.2**	90.8	**77.9**	66.5	**72.8**
~7B Model Size								
MMRet (Zhou et al., 2025)	LLaVA-1.6-7B	56.0	57.4	69.9	83.6	68.0	59.1	64.1
IDMR (Liu et al., 2025a)	InternVL2.5-8B	58.3	58.6	68.7	85.6	70.5	57.9	64.9
VLM2Vec (Jiang et al., 2025)	Qwen2VL-7B	62.8	56.5	69.4	81.9	71.3	57.9	65.4
UniME (Gu et al., 2025)	LLaVA-1.6-7B	60.6	52.9	67.9	85.1	68.4	57.9	66.6
CAFe (Yu et al., 2025)	LLaVA-OneVision-7B	65.2	65.6	70.0	91.2	75.8	62.4	69.8
UNITE (Kong et al., 2025)	Qwen2VL-7B	68.3	65.1	71.6	84.8	73.6	66.3	70.3
LLaVE (Lan et al., 2025)	LLaVA-NeXT-7B	65.7	65.4	70.9	91.9	75.0	64.4	70.3
UME-R1 (Lan et al., 2026)	Qwen2VL-7B	67.1	69.2	71.9	84.9	76.1	65.1	71.2
ALICE (ours)	Qwen2-VL-7B	**71.8**	70.3	**75.8**	90.6	**79.6**	**68.8**	**74.8**
ALICE (ours)	Qwen2.5VL-7B	68.5	**71.2**	73.7	90.4	78.8	66.6	73.4
ALICE (ours)	Qwen3VL-8B	70.4	70.5	74.5	**92.4**	79.3	68.0	74.2

4.3 Ablation Studies

All ablations are based on Qwen2.5VL-3B using $K = 15$ latent tokens as default setting.

Inference efficiency. Table 2 compares per-query latency on a single H20 GPU using the CIRR (Liu et al., 2021) subset, which requires compositional reasoning over multimodal queries (image and text). ALICE achieves the highest accuracy while adding only 1.4 ms over VLM2Vec (6.7 vs. 5.3 ms), and is orders of magnitude faster than RGE (Liu et al., 2025b) (2,018 ms) and UME-R1 (Lan et al., 2026) (7,229 ms), which autoregressively generate tens to hundreds of tokens per query.

Table 2: **Inference latency and Hit@1 on CIRR (MMEB).** Latency and generated tokens are measured per query.

	Generated tokens	Latency (ms/query)	Hit@1
VLM2Vec	0	**5.3** $\pm$ 3.2	46.8
UME-R1	251.5 $\pm$ 177.8	7229.0 $\pm$ 5109.1	53.7
RGE	53.4 $\pm$ 8.2	2018.0 $\pm$ 299.2	54.0
ALICE (ours)	0	6.7 $\pm$ 3.4	**58.0**

Effect of each training stage. Table 3 disentangles each training stage. The first row shows our VLM2Vec-style baseline, which applies only free-form contrastive training and extracts the final `<emb>` token as the embedding. Appending latent tokens without CoT alignment yields only a marginal gain (+0.2), showing that adding latent positions alone provides little benefit in this setting. CoT alignment produces a substantial jump (+3.9), already achieving competitive performance among 2B–4B models. Free-form training adds +2.2 by allowing adaptation beyond rigid step-level anchors.

Alignment objective design. Table 4 ablates the training loss during the CoT alignment stage. Removing the final-embedding contrastive loss $\mathcal{L}_{\text{emb}}$ reduces the overall score from 71.9 to 71.1,

Table 3: **Ablation of each training stage.**

w/ latent	CoT Alignment	Free-form Training	Per Meta-Task Score				IND	OOD	Overall
			CLS	VQA	Retrieval	Grounding			
✗	✗	✓	58.4	61.9	67.5	89.1	72.0	58.1	65.8
✓	✗	✓	59.7	61.4	67.0	89.9	72.0	58.5	66.0
✓	✓	✗	65.0	66.8	69.6	89.1	74.5	63.7	69.7
✓	✓	✓	66.8	68.7	72.4	90.6	77.0	65.4	71.9

Table 4: **Ablation on the training objective in the CoT alignment stage.**

Training Objective	Per Meta-Task Score				IND	OOD	Overall
	CLS	VQA	Retrieval	Grounding			
$L_{emb} + L_{align}$(stepwise) (ALICE)	66.8	68.7	72.4	90.6	77.0	65.4	71.9
L_{align} (stepwise) only	67.0	67.8	71.1	89.4	76.3	64.6	71.1
$L_{emb} + L_{align}$(stepwise cosine loss)	65.7	67.7	71.3	87.3	76.2	63.4	70.5
$L_{emb} + L_{align}$(global)	64.9	66.8	69.9	89.2	74.4	64.0	69.8

Table 5: **Ablation on the alignment target in the CoT alignment stage.**

Alignment Target	Per Meta-Task Score				IND	OOD	Overall
	CLS	VQA	Retrieval	Grounding			
w/o Alignment	59.7	61.4	67.0	89.9	72.0	58.5	66.0
Vocabulary Embedding	63.8	64.1	69.6	89.3	73.9	62.1	68.6
Target Embedding	64.8	67.3	70.7	89.8	75.0	64.2	70.2
ALICE (ours)	66.8	68.7	72.4	90.6	77.0	65.4	71.9

supporting direct retrieval supervision of <emb> during alignment. Replacing the stepwise InfoNCE loss with a stepwise cosine similarity loss also hurts, as the absence of in-batch negatives may lead to representation collapse. Finally, using a single global grouping over the entire rationale instead of per-step anchors reduces performance as well, confirming the value of step-level granularity when aligning latent tokens with CoT anchors.

Alignment target. Table 5 compares four alignment targets: (i) no alignment, using <latent> tokens without any supervision; (ii) stepwise alignment of <latent> hidden states with the mean-pooled hidden states of each explicit CoT step (Vocabulary Embedding); (iii) stepwise alignment with the target embeddings (Target Embedding); and (iv) our per-step CoT anchors from the reference embedder (ALICE). Option (ii) improves over the unaligned baseline, demonstrating the importance of latent alignment; however, it operates in the vocabulary embedding space rather than the semantic embedding space, limiting its effectiveness. Directly aligning with target embeddings (iii) also yields gains through repeated emphasis on the target representation, but lacks structural decomposition across reasoning steps. Our per-step CoT anchors perform best, indicating that step-level granularity and grounding in semantic embedding space are essential for latent alignment.

Effect of latent token budget. We vary latent tokens per step from one to four with $T = 5$ reasoning steps. ALICE consistently outperforms the COCONUT-like baseline by 2.2–2.9 points across all tested configurations (Fig. 4), using a single forward pass. This advantage supports our motivation in Sec. 1: direct embedding-space alignment provides richer semantic supervision for retrieval than a next-token prediction objective biased toward local token prediction. Performance improves modestly from one to three tokens per step, then decreases slightly at four, suggesting that excessive latent capacity may introduce redundancy or noise rather than additional useful reasoning information, potentially diluting the alignment signal. We default to three tokens per step ($K = 15$).

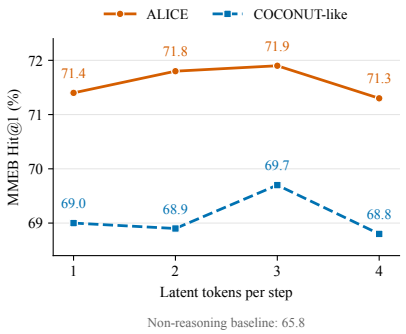
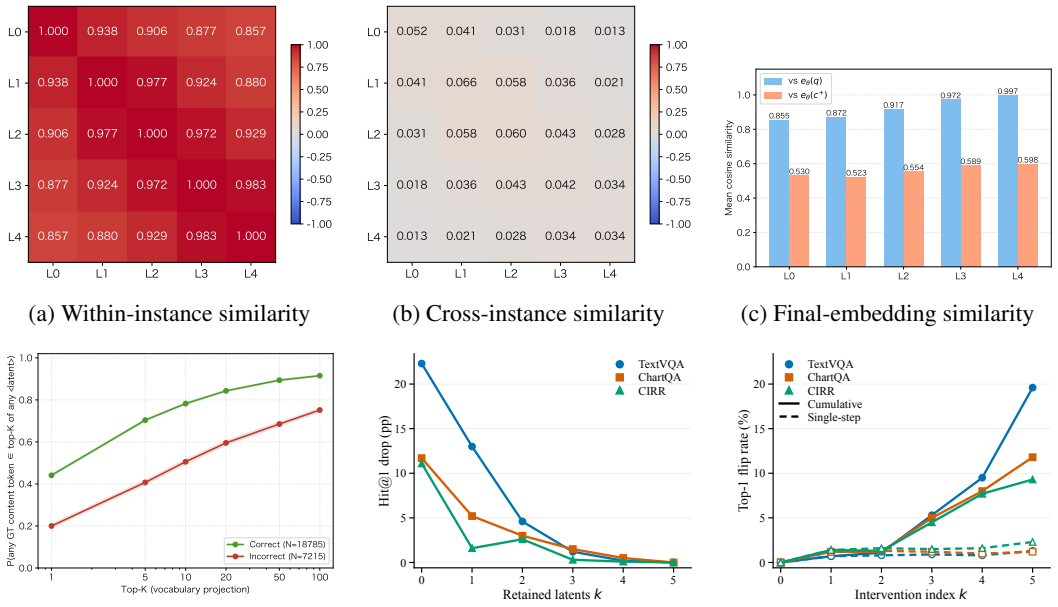


Figure 4: **ALICE vs. COCONUT-like training on MMEB.**

4.4 Analysis of the <latent> Tokens

To better understand how ALICE's latent tokens behave, we analyze the $K = 5$ variant through representation similarity, answer-token presence, and early-stop/intervention analyses (Fig. 5).

Representation similarity. Motivated by the similarity diagnostics of Li et al. (2026a), we examine the geometry of the latent states. Within each instance, nearby latent positions are more similar than distant ones, while corresponding positions across instances have low similarity. Later latents also become more aligned with the final embedding (Fig. 5a–c). Together, these patterns suggest instance-specific representations that evolve across latent positions toward the retrieval output.



(a) Within-instance similarity (b) Cross-instance similarity (c) Final-embedding similarity

(d) Answer-token presence (e) Early stopping (f) KV-cache interventions

Figure 5: **Analysis of the `<latent>` tokens** (K=5). Cosine similarity (a) between latent positions within an instance, (b) at corresponding positions across instances, and (c) between latents and the final embedding. (d) Answer-token presence among top-ranked vocabulary projections, grouped by retrieval correctness. (e) Hit@1 drop relative to all five latents when <emb> follows the first k retained latents. (f) Top-1 prediction flip rate after zeroing the KV-cache of $L_1, \ldots, L_k$ (solid) or L_k alone (dashed), while preserving subsequent computation. In (f), $k = 0$ denotes no intervention.

Answer-token presence. Adapting the vocabulary-projection analysis of Dilgren & Wiegreffe (2026), we measure whether tokens from textual retrieval targets occur among the top-ranked vocabulary predictions of the latent states, excluding image-only targets. Correctly retrieved instances show higher answer-token presence than incorrect ones (Fig. 5d), linking successful retrieval to answer-relevant information in the latents.

Early stopping and latent interventions. Inspired by Li et al. (2026b), we examine when latent computation becomes sufficient for retrieval and how perturbing earlier latents affects subsequent predictions. For early stopping, we retain the first k latents and place <emb> immediately after them, omitting the remaining latents. Most gains emerge within the first three positions, with smaller improvements thereafter (Fig. 5e). For intervention, we preserve sequence length and positions but zero the KV-cache entries of L_k or $L_1, \ldots, L_k$ during incremental encoding, before computing subsequent latents and <emb>. Single-position interventions change at most 2.3% of top-1 predictions, whereas cumulative intervention over all five positions changes 19.6%, 11.8%, and 9.3% on TextVQA, ChartQA, and CIRR, respectively (Fig. 5f). The same cumulative interventions reduce Hit@1 by 18.8, 10.4, and 7.7 percentage points, respectively. Together, these results suggest that retrieval-relevant information develops across latent positions and contributes to the final decision.

5 Conclusion

We presented ALICE, a semantic-alignment approach to latent reasoning for multimodal embedding without autoregressive inference. Observing that autoregressive curricula may bias latent representations toward local next-token prediction rather than holistic reasoning semantics, we instead align latent hidden states directly with per-step CoT anchors in the semantic embedding space via a two-phase pipeline—CoT-anchored alignment followed by free-form contrastive adaptation. On MMEB, ALICE achieves strong results across the 2B–8B scale, reaching 74.8 with Qwen2-VL-7B, with minimal additional latency. Our analyses reveal structured, instance-specific latent representations that contribute to retrieval decisions, suggesting reasoning-like behavior in latent space.

REFERENCES

Haonan Chen, Hong Liu, Yuping Luo, Liang Wang, Nan Yang, Furu Wei, and Zhicheng Dou. MoCa: Modality-aware continual pre-training makes better bidirectional multimodal embeddings. *arXiv preprint arXiv:2506.23115*, 2025.

Haonan Chen, Chu Li, Zhicheng Wang, Yuanwei Liu, Yuanjiang Wang, Shaohua Jiang, and Zhicheng Dou. Douyin multimodal embedding model technical report, 2026. URL https://arxiv.org/abs/2608.02148.

Xuanming Cui, Jianpeng Cheng, Hong-You Chen, Satya Narayan Shukla, Abhijeet Awasthi, Xichen Pan, Chaitanya Ahuja, Shlok Kumar Mishra, Taipeng Tian, Qi Guo, Ser-Nam Lim, Aashu Singh, and Xiangjun Fan. Think then embed: Generative context improves multimodal embedding. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=AKXXMK5YTI.

Connor Dilgren and Sarah Wiegreffe. Are latent reasoning models easily interpretable? In *Workshop on Latent & Implicit Thinking – Going Beyond CoT Reasoning*, 2026. URL https://openreview.net/forum?id=L4k8rbmwrr.

Manuel Faysse, Hugues Sibille, Tony Wu, Bilel Omrani, Gautier Viaud, Céline Hudelot, and Pierre Colombo. ColPali: Efficient document retrieval with vision language models. In *International Conference on Learning Representations*, volume 2025, pp. 61424–61449, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/99e9e141aafc314f76b0ca3dd66898b3-Paper-Conference.pdf.

Sachin Goyal, Ziwei Ji, Ankit Singh Rawat, Aditya Menon, Sanjiv Kumar, and Vaishnavh Nagarajan. Think before you speak: Training language models with pause tokens. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=ph04CRkPdC.

Tiancheng Gu, Kaicheng Yang, Ziyong Feng, Xingjun Wang, Yanzhao Zhang, Dingkun Long, Yingda Chen, Weidong Cai, and Jiankang Deng. Breaking the modality barrier: Universal embedding learning with multimodal LLMs. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 2860–2869, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason E Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. In *Workshop on Reasoning and Planning for Large Language Models*, 2025. URL https://openreview.net/forum?id=KrWSrrYGpT.

Xiangzhao Hao, Shijie Wang, Tianyu Yang, Tianyue Wang, Haiyun Guo, and Jinqiao Wang. TRACE: Task-adaptive reasoning and representation learning for universal multimodal retrieval. *arXiv preprint arXiv:2603.02929*, 2026.

Chenwei He, Xiangzhao Hao, Tianyu Yang, Yuxiang Ma, Yuheng Jia, Lingxiang Wu, Chaoyang Zhao, Haiyun Guo, and Jinqiao Wang. Plume: Latent reasoning based universal multimodal embedding, 2026. URL https://arxiv.org/abs/2604.02073.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9729–9738, 2020.

Jui-Ting Huang, Ashish Sharma, Shuying Sun, Li Xia, David Zhang, Philip Pronin, Janani Padmanabhan, Giuseppe Ottaviano, and Linjun Yang. Embedding-based retrieval in Facebook search. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2553–2561, 2020.

Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Yao Hu, and Shaohui Lin. Vision-R1: Incentivizing reasoning capability in multimodal large language models. *arXiv preprint arXiv:2503.06749*, 2025.

Haonan Jiang, Yuji Wang, Yongjie Zhu, Xin Lu, Wenyu Qin, Meng Wang, Pengfei Wan, and Yansong Tang. Embed-RL: Reinforcement learning for reasoning-driven multimodal embeddings. *arXiv preprint arXiv:2602.13823*, 2026.

Ting Jiang, Minghui Song, Zihan Zhang, Haizhen Huang, Weiwei Deng, Feng Sun, Qi Zhang, Deqing Wang, and Fuzhen Zhuang. E5-v: Universal embeddings with multimodal large language models. *arXiv preprint arXiv:2407.12580*, 2024.

Ziyan Jiang, Rui Meng, Xinyi Yang, Semih Yavuz, Yingbo Zhou, and Wenhu Chen. VLM2vec: Training vision-language models for massive multimodal embedding tasks. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=TE0KOzWYAF.

Zixuan Ke, Fangkai Jiao, Yifei Ming, Xuan-Phi Nguyen, Austin Xu, Do Xuan Long, Minzhi Li, Chengwei Qin, PeiFeng Wang, Silvio Savarese, Caiming Xiong, and Shafiq Joty. A survey of frontiers in LLM reasoning: Inference scaling, learning to reason, and agentic systems. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=SlsZZ25InC. Survey Certification.

Fanheng Kong, Jingyuan Zhang, Yahui Liu, Hongzhi Zhang, Shi Feng, Xiaocui Yang, Daling Wang, Yu Tian, Fuzheng Zhang, Guorui Zhou, et al. Modality curation: Building universal embeddings for advanced multimodal information retrieval. *arXiv preprint arXiv:2505.19650*, 2025.

Zhibin Lan, Liqiang Niu, Fandong Meng, Jie Zhou, and Jinsong Su. LLaVE: Large language and vision embedding models with hardness-weighted contrastive learning. *arXiv preprint arXiv:2503.04812*, 2025.

Zhibin Lan, Liqiang Niu, Fandong Meng, Jie Zhou, and Jinsong Su. UME-r1: Exploring reasoning-driven generative multimodal embeddings. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=2ius36JQUJ.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in neural information processing systems*, 33: 9459–9474, 2020.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pp. 12888–12900. PMLR, 2022.

You Li, Chi Chen, Yanghao Li, Fanhu Zeng, Kaiyu Huang, Jinan Xu, and Maosong Sun. Imagination helps visual reasoning, but not yet in latent space. *arXiv preprint arXiv:2602.22766*, 2026a.

Zirui Li, Xuefeng Bai, Kehai Chen, Yizhi Li, Jian Yang, Chenghua Lin, and Min Zhang. Dynamics within latent chain-of-thought: An empirical study of causal structure. In *Forty-third International Conference on Machine Learning*, Seoul, South Korea, 2026b. URL https://openreview.net/forum?id=kHB8m3ojGe.

Bangwei Liu, Yicheng Bao, Shaohui Lin, Xuhong Wang, Xin Tan, Yingchun Wang, Yuan Xie, and Chaochao Lu. IDMR: Towards instance-driven precise visual correspondence in multimodal retrieval. *arXiv preprint arXiv:2504.00954*, 2025a.

Chunxu Liu, Jiyuan Yang, Ruopeng Gao, Yuhan Zhu, Feng Zhu, Rui Zhao, and Limin Wang. Reasoning guided embeddings: Leveraging MLLM reasoning for improved multimodal retrieval. *arXiv preprint arXiv:2511.16150*, 2025b.

Yikun Liu, Yajie Zhang, Jiayin Cai, Xiaolong Jiang, Yao Hu, Jiangchao Yao, Yanfeng Wang, and Weidi Xie. Lamra: Large multimodal model as your advanced retrieval assistant. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 4015–4025, 2025c.

Zheyuan Liu, Cristian Rodriguez-Opazo, Damien Teney, and Stephen Gould. Image retrieval on real-life images with pre-trained vision-and-language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2125–2134, October 2021.

Rui Meng, Ziyan Jiang, Ye Liu, Mingyi Su, Xinyi Yang, Yuepeng Fu, Can Qin, Raghuveer Thirukovalluru, Xuan Zhang, Zeyuan Chen, Ran Xu, Caiming Xiong, Yingbo Zhou, Wenhu Chen, and Semih Yavuz. VLM2vec-v2: Advancing multimodal embedding for videos, images, and visual documents. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=TpU38jbKIJ.

Bhaskar Mitra, Fernando Diaz, and Nick Craswell. Learning to match using local and distributed representations of text for web search. In *Proceedings of the 26th international conference on world wide web*, pp. 1291–1299, 2017.

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.

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.

Zhenyi Shen, Hanqi Yan, Linhai Zhang, Zhanghao Hu, Yali Du, and Yulan He. CODI: Compressing chain-of-thought into continuous space via self-distillation. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 677–693, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.36. URL https://aclanthology.org/2025.emnlp-main.36/.

Aditi Singh, Abul Ehtesham, Saket Kumar, and Tala Talaei Khoei. Agentic retrieval-augmented generation: A survey on agentic RAG. *arXiv preprint arXiv:2501.09136*, 2025.

Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. EVA-CLIP: Improved training techniques for CLIP at scale. *arXiv preprint arXiv:2303.15389*, 2023.

Cong Wei, Yang Chen, Haonan Chen, Hexiang Hu, Ge Zhang, Jie Fu, Alan Ritter, and Wenhu Chen. UniIR: Training and benchmarking universal multimodal information retrievers. In *European Conference on Computer Vision*, pp. 387–404. Springer, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Xilin Wei, Xiaoran Liu, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Jiaqi Wang, Xipeng Qiu, and Dahua Lin. SIM-COT: Supervised implicit chain-of-thought. In *International Conference on Learning Representations*, 2026.

Hao Yu, Zhuokai Zhao, Shen Yan, Lukasz Korycki, Jianyu Wang, Baosheng He, Jiayi Liu, Lizhu Zhang, Xiangjun Fan, and Hanchao Yu. CAFe: Unifying representation and generation with contrastive-autoregressive finetuning. *arXiv preprint arXiv:2503.19900*, 2025.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 11975–11986, 2023.

Kai Zhang, Yi Luan, Hexiang Hu, Kenton Lee, Siyuan Qiao, Wenhu Chen, Yu Su, and Ming-Wei Chang. MagicLens: Self-supervised image retrieval with open-ended instructions. *arXiv preprint arXiv:2403.19651*, 2024.

Xin Zhang, Yanzhao Zhang, Wen Xie, Mingxin Li, Ziqi Dai, Dingkun Long, Pengjun Xie, Meishan Zhang, Wenjie Li, and Min Zhang. Bridging modalities: Improving universal multimodal retrieval by multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 9274–9285, 2025.

Junjie Zhou, Yongping Xiong, Zheng Liu, Ze Liu, Shitao Xiao, Yueze Wang, Bo Zhao, Chen Jason Zhang, and Defu Lian. MegaPairs: Massive data synthesis for universal multimodal retrieval. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 19076–19095, 2025.

Yuhan Zhu, Xiangyu Zeng, Chenting Wang, Xinhao Li, Chunxu Liu, Yicheng Xu, Ziang Yan, Yi Wang, and Limin Wang. FreeRet: MLLMs as training-free retrievers. *arXiv preprint arXiv:2509.24621*, 2026.

A LATENT-TOKEN DIAGNOSTICS ON PLUME

To examine whether the first-token concentration observed in Fig. 1 also occurs in PLUME (He et al., 2026), we project its latent hidden states onto the vocabulary and inspect their top-ranked predictions, following the coverage analysis in Fig. 2. PLUME combines a COCONUT-style curriculum with an external MoE adapter, providing a second setting in which to examine how next-token supervision shapes latent representations.

We follow PLUME's own segmentation of its CoT annotations and collect the 500 most frequent step-initial tokens, covering 93.2% of step-initial occurrences. PLUME retains a next-token prediction loss in the final curriculum stage, where predictions also include sequence-ending markers. We therefore add <gen_emb>, </, and <eot> to the reference set to capture this ending-token behavior alongside step-initial predictions. The resulting set covers 58.4% of the latent states' top-5 vocabulary predictions (88.3% for top-3 predictions).

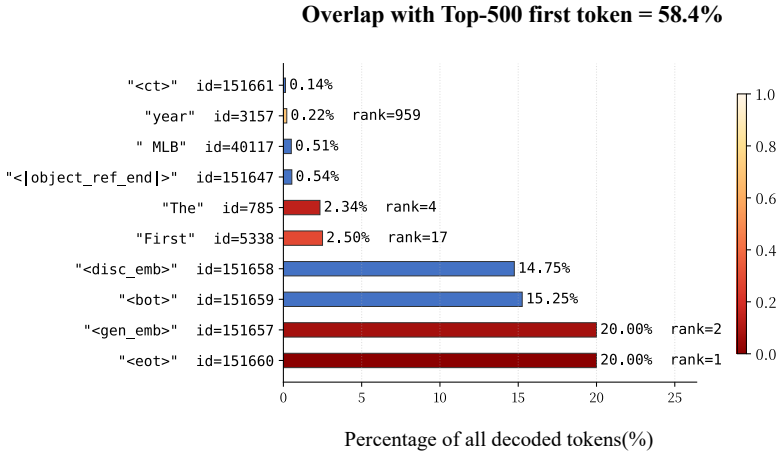


Figure 6: **Vocabulary projections of PLUME's latent states.** Token frequencies among top-5 predictions show concentration on step-initial words and special tokens.

Fig. 6 also reveals substantial mass on <bot> and <disc_emb>, accounting for 15.25% and 14.75% of top-5 predictions. These two special tokens lie outside the expected prediction targets. Their prominence suggests a drift from the ending markers <eot> and <gen_emb> toward other special tokens. Including them brings the combined coverage to approximately 88%, showing that the concentration extends from CoT step-initial words to a small special-token vocabulary.

PLUME's component ablations help relate this behavior to retrieval performance (Table 6). Removing the MoE adapter reduces the overall score from 61.6 to 59.2, accounting for 2.4 of the 2.8-point gain over the configuration without latent transition (58.8). The single-MLP variant retains only a 0.4-point gain and scores below the explicit-CoT embedder UME-R1 (60.1) on MMEB-v2. Together with the vocabulary projections, these results suggest that a next-token-based latent curriculum alone provides limited retrieval gains, motivating ALICE's direct alignment with CoT semantics in embedding space.

Table 6: **Selected PLUME component ablations on MMEB-v2**, reported by He et al. (2026). All averages over the 78 datasets.

Configuration	Image	Video	VisDoc	All
w/o Latent Transition	63.6	41.0	64.8	58.8
w/o MoE (single MLP)	64.2	41.8	64.4	59.2
Full PLUME	66.3	44.1	67.5	61.6

B EVALUATION ON MMEB-V2

We extend the same CoT annotation and refinement pipeline described in Sec. 3.2 to the video and visual document training data in MMEB-v2. The additional data cover video caption retrieval, video question answering, text-to-video retrieval, and the VisRAG and ViDoRe-ColPali tasks. Together with the image-training subsets, this yields 1.8M CoT-annotated training examples. We train ALICE on this expanded dataset using the same three-stage procedure.

We compare ALICE with ColPali (Faysse et al., 2025), VLM2Vec (Jiang et al., 2025), VLM2Vec-V2 (Meng et al., 2026), LamRA (Liu et al., 2025c), and CAFe (Yu et al., 2025), as well as the reasoning-based embedders UME-R1 (Lan et al., 2026), TTE (Cui et al., 2026), Embed-RL (Jiang et al., 2026), and PLUME (He et al., 2026). Table 7 reports results across seven ALICE backbones on the 78 image, video, and visual document datasets of MMEB-v2. We additionally provide results for closed-source methods or methods trained with additional data as references. On MMEB-v2, ALICE achieves competitive overall performance without autoregressive reasoning at inference. With Qwen3-VL-4B, it scores 67.6 overall versus Embed-RL's 68.1, with stronger image and visual-document results but lower video performance. The larger gaps in video retrieval and moment retrieval suggest that temporal evidence selection and integration remain important directions for improving our fixed-budget latent reasoning approach.

Table 7: **Results on MMEB-v2.** Within the 2B–4B and 7B–8B groups, the best and second-best scores in each column are marked in **bold** and underline, respectively. Image and video tasks use Hit@1; visual document tasks use nDCG@5. Avg. and All denote dataset-level averages within each modality and across all 78 datasets, respectively. CLS: classification; QA: question answering; RET: retrieval; GD: grounding; MRET: moment retrieval; VDR: ViDoRe; VR: VisRAG; OOD: out-of-distribution visual document tasks.

Model	Image					Video					VisDoc					All
	CLS	QA	RET	GD	Avg.	CLS	QA	RET	MRET	Avg.	VDRv1	VDRv2	VR	OOD	Avg.	
# Datasets	10	10	12	4	36	5	5	5	3	18	10	4	6	4	24	78
Closed-Source or w/ Additional Data (Reference)																
GME (Qwen2-VL-7B)	57.7	34.7	71.2	59.3	56.0	37.4	50.4	28.4	38.2	38.6	89.4	55.6	85.0	44.4	75.2	57.8
Ops-MM-Embed (Qwen2-VL-7B)	69.7	69.6	73.1	87.2	72.7	59.7	62.2	45.7	43.2	53.8	80.1	59.6	79.3	67.8	74.4	68.9
RzenEmbed-v1 (Qwen2-VL-7B)	69.8	68.7	76.8	85.7	73.6	52.8	56.2	41.9	41.8	48.9	89.5	60.8	87.9	44.4	76.8	68.9
Qwen3-VL-Embedding (Qwen3-VL-8B)	74.2	81.1	80.2	92.3	80.1	78.4	71.0	58.7	56.1	67.1	87.2	69.9	88.7	73.3	82.4	77.8
2B–4B Model Size																
ColPali-V1.3 (PaliGemma-3B)	40.3	11.5	48.1	40.3	34.9	26.7	37.8	21.6	25.5	28.2	**83.6**	52.0	81.1	43.1	71.0	44.4
VLM2Vec (Qwen2-VL-2B)	58.7	49.3	65.0	72.9	59.7	33.4	30.5	20.6	33.0	29.0	49.8	13.5	51.8	33.5	41.6	47.0
VLM2Vec-V2 (Qwen2-VL-2B)	62.9	56.3	69.5	77.3	64.9	39.3	34.3	28.8	36.8	34.6	75.5	44.9	79.4	39.4	65.4	58.0
UME-R1 (Qwen2-VL-2B)	64.8	62.8	67.6	77.2	66.6	44.3	51.2	32.9	39.7	42.2	72.4	46.2	79.2	37.2	63.9	60.1
TTE_s (Qwen3-VL-2B)	67.9	66.6	70.2	84.1	70.1	47.3	49.1	33.2	32.1	41.3	77.5	53.2	83.2	41.1	68.8	63.1
Embed-RL (Qwen3-VL-2B)	62.8	67.9	68.6	90.4	69.2	57.0	55.9	**45.1**	49.4	52.1	79.9	52.0	84.6	65.7	74.1	66.8
Embed-RL (Qwen3-VL-4B)	63.7	**70.5**	71.3	91.4	71.2	**57.6**	**58.4**	**45.1**	**49.5**	**53.0**	80.2	53.4	84.9	**67.1**	74.7	**68.1**
PLUME (Qwen2-VL-2B)	66.5	59.2	67.6	79.7	66.3	45.0	52.3	33.5	46.7	44.1	72.1	49.8	78.1	57.4	67.5	61.6
ALICE (Qwen2-VL-2B)	**69.2**	65.3	71.9	88.0	71.1	51.6	48.0	33.2	34.6	42.7	78.1	48.6	84.9	64.2	72.6	65.0
ALICE (Qwen2.5-VL-3B)	66.3	69.2	72.1	89.3	71.6	48.1	55.4	32.6	35.0	43.7	80.3	**54.5**	85.1	66.7	74.9	66.2
ALICE (Qwen3-VL-2B)	68.6	65.6	71.3	90.3	71.1	54.1	51.4	35.5	38.2	45.5	79.0	54.3	85.6	66.5	74.5	66.2
ALICE (Qwen3-VL-4B)	68.6	68.6	**73.3**	**91.6**	**72.7**	53.8	53.2	36.7	41.9	46.9	81.4	54.1	**85.8**	66.9	**75.5**	67.6
7B–8B Model Size																
LamRA (Qwen2-VL-7B)	59.2	26.5	70.0	62.7	54.1	39.3	42.6	24.3	34.6	35.2	22.0	11.5	37.4	21.0	23.9	40.4
VLM2Vec (Qwen2-VL-7B)	62.7	56.9	69.4	82.2	65.5	39.1	30.0	29.0	40.6	34.0	56.9	9.4	59.1	38.1	46.4	52.3
VLM2Vec-V2 (Qwen2-VL-7B)	65.7	61.5	70.0	85.2	68.1	45.9	33.9	27.6	39.3	36.4	78.8	52.6	82.7	42.1	69.3	61.2
CAFe (LLaVA-OV-7B)	63.6	61.7	69.1	87.6	67.6	35.8	58.7	34.4	39.5	42.4	70.7	49.6	79.5	38.1	63.9	60.6
UME-R1 (Qwen2-VL-7B)	67.1	69.2	71.9	84.9	71.3	48.6	**60.7**	**38.2**	39.3	**47.5**	75.7	50.5	83.7	37.6	67.1	64.5
TTE_s (Qwen2-VL-7B)	69.7	**72.4**	74.0	90.6	74.2	49.1	60.6	36.4	37.2	46.8	**84.1**	**62.7**	**91.9**	47.6	76.4	68.6
ALICE (Qwen2-VL-7B)	**71.8**	70.3	**75.8**	90.6	**74.8**	**56.3**	53.7	35.4	36.9	46.5	81.9	53.4	88.1	67.3	76.3	68.7
ALICE (Qwen2.5-VL-7B)	68.5	71.3	73.8	90.7	73.5	51.2	57.5	34.2	39.7	46.3	81.4	52.6	87.5	67.5	75.8	67.9
ALICE (Qwen3-VL-8B)	69.8	71.5	74.6	**92.3**	74.4	54.6	54.9	35.1	**43.8**	**47.5**	82.4	54.4	87.8	**68.0**	**76.7**	**68.9**

C CoT Supervision Structure

We compare alignment supervision on a balanced subset of 160K training examples (8K from each of the 20 in-domain datasets), using Qwen2.5-VL-3B with 15 latent tokens and keeping the other training settings fixed. Table 8 reports overall Hit@1 on MMEB.

Table 8: **Ablation of CoT supervision structure.**

Alignment supervision	Overall
Supervision controls	
No alignment (15 latent tokens)	63.9
Task specification only	64.0
CoT from other instances	63.0
Single-anchor alignment	
Direct target embedding	66.1
Global CoT embedding	66.0
Stepwise alignment	
Shuffled CoT steps	66.7
Original CoT steps (ALICE)	**67.9**

The supervision controls indicate that useful alignment depends on instance-specific content: generic task descriptions provide little gain, while mismatched CoTs degrade performance. Direct target and global CoT alignment both improve over the unaligned baseline, showing the value of target-relevant semantic supervision. Both use a shared anchor across latent groups, derived from either the target or the full CoT. ALICE gains a further 1.8 and 1.9 points, respectively, suggesting that distinct step-level anchors help transfer diverse, complementary semantic information into the latents.

Shuffled CoT supervision retains substantial gains over the unaligned baseline, suggesting that complementary step-level content remains useful after reordering. Preserving the original step correspondence further improves performance by 1.2 points, suggesting the value of organizing this content into distinct, corresponding latent groups.

D Training Cost

For Qwen2.5-VL-3B with $K = 5$, ALICE's reference-embedder training, latent alignment, and free-form contrastive training take $1.5\,\mathrm{h} \times 16$, $5\,\mathrm{h} \times 32$, and $4\,\mathrm{h} \times 32$ GPUs, respectively, totaling 312 GPU-hours for model training. Our COCONUT-like implementation takes approximately 1,350 GPU-hours across five curriculum epochs, about $4.3\times$ the training compute of ALICE.

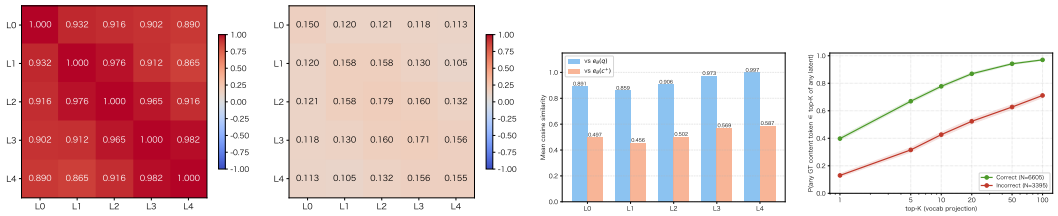
E TASK-SPECIFIC ANALYSIS OF LATENT TOKENS

Following the aggregated analysis in Sec. 4.4, we break down the latent-token behavior by individual meta-task. All results use the $K{=}5$ variant for clearer visualization. Figures 7–10 present four panels per task: (a) intra-instance pairwise similarity, (b) cross-instance similarity at corresponding positions, (c) cosine similarity between each `<latent>` and the final `<emb>`, and (d) answer-token hit rate (or, for visual grounding, a 2-D projection of the latent representations).

Consistent intra- and cross-instance patterns. Across all four tasks, the intra-instance similarity (panels a) decreases with positional distance, and the cross-instance similarity (panels b) remains uniformly low. This confirms that the sequential, instance-specific structure observed in the aggregated analysis holds at the per-task level.

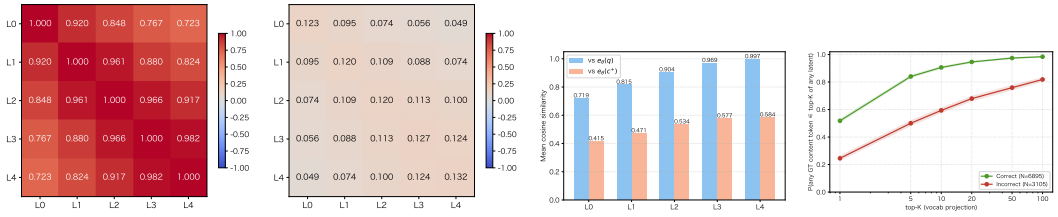
Task-dependent convergence behavior. For classification (Fig. 7c) and VQA (Fig. 8c), the cosine similarity between latent tokens and the final embedding shows a clear monotonic increase across positions, indicating progressive convergence toward the output representation. We hypothesize that these two tasks most closely resemble the MLLM's native reasoning behavior, where the model refines its understanding step by step toward a final answer. In contrast, retrieval (Fig. 9c) and visual grounding (Fig. 10c) exhibit a relatively flat similarity profile. We attribute this to the nature of these tasks: the reasoning process tends to produce descriptive characterizations of the query rather than a progressive refinement toward a single target concept.

Answer-token hit rate. For classification, VQA, and retrieval tasks with textual targets (combining VisualNews_i2t, MSCOCO_i2t, WebQA, OVEN, and EDIS), we report the distribution of answer-token hit rates in panels (d). Consistent with the aggregated analysis, correctly retrieved instances exhibit higher answer-token presence in the latent decoding results across all three task types. For visual grounding, only the RefCOCO-Matching subset contains textual targets, and the answer-token hit rate shows little separation between correct and incorrect instances (Fig. 10d). We attribute this to the symmetric nature of the subset: both queries and targets are short referring expressions with similar descriptive vocabulary, making the hit-rate metric less discriminative for this particular task.



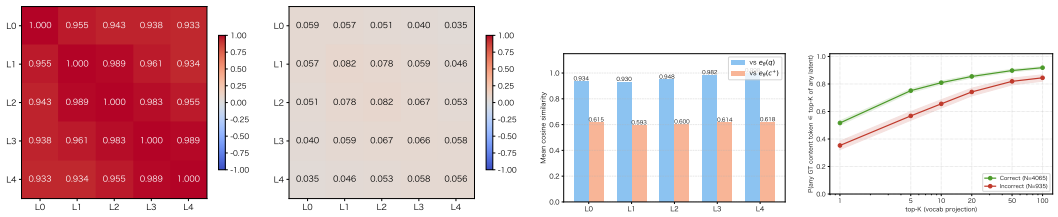
(a) Intra-instance (b) Cross-instance (c) Similarity to `<emb>` (d) Answer-token hit rate

Figure 7: **Latent token analysis – Classification.**



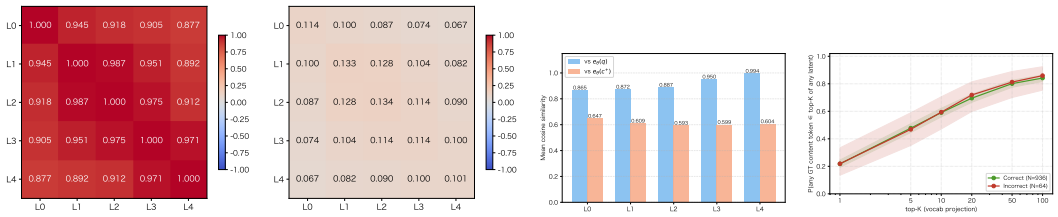
(a) Intra-instance (b) Cross-instance (c) Similarity to `<emb>` (d) Answer-token hit rate

Figure 8: **Latent token analysis – VQA.**



(a) Intra-instance (b) Cross-instance (c) Similarity to <emb> (d) Answer-token hit rate

Figure 9: **Latent token analysis – Retrieval.**



(a) Intra-instance (b) Cross-instance (c) Similarity to <emb> (d) Answer-token hit rate

Figure 10: **Latent token analysis – Visual Grounding.**

F Limitations and Future Work

ALICE transfers stepwise CoT semantics into latent representations, yielding effective retrieval with minimal additional latency. While our analyses suggest reasoning-like behavior, distinguishing the integration of complementary semantic information from compositional reasoning remains an open question for latent reasoning methods, whether trained autoregressively or through semantic alignment. These mechanisms may also coexist and contribute differently across tasks. Future work could investigate how task complexity and evidence dependencies shape the benefits of latent reasoning, and how to better organize complementary information within the latent space for tasks requiring multi-step inference.

G AI Assistance Declaration

We used AI assistance to polish the language of this manuscript. We also used `Qwen3VL-235B-A22B-Instruct` to generate the CoT annotations for our training data, as described in Sec. 3.2.

H Reproducibility Statement

Section 3 describes the model architecture and training objectives, and Sec. 4.1 specifies the experimental settings and training hyperparameters. Appendix I provides the CoT generation and refinement prompts and training-data statistics. We will publicly release all code and training data upon acceptance.

I STEPWISE COT DATA CONSTRUCTION DETAILS

This appendix provides the full prompts used for constructing and filtering the stepwise chain-of-thought (CoT) rationales described in Sec. 3.2. Template variables are shown in {{ }} notation. We provide the prompt of CIRR subset for example.

After the annotation and refinement process, we manually audited 2,000 query–rationale–target tuples, with 100 examples from each of the 20 MMEB training subsets. We found no explicit target references or target-only attributes in the audited samples.

I.1 COT STEP AND LENGTH DISTRIBUTIONS

The CoT annotations for the 20 MMEB image-training subsets comprise 1,031,523 rationales. Table 9 summarizes their step counts and text lengths. Five-step rationales account for 67.90% of the data, followed by four-step (19.84%) and six-step (12.02%) rationales. A rationale contains 84.5 words on average, with 17.2 words per step. Length is measured in whitespace-separated words over the CoT text.

Table 9: **Step-count and length distributions of the training CoTs.** P10 and P90 denote the 10th and 90th percentiles.

Number of steps	3	4	5	6	Other
CoTs	2,094	204,698	700,444	123,964	323
Share (%)	0.20	19.84	67.90	12.02	0.03

Length (words)	Mean	Median	P10	P90
Per CoT	84.5	84.0	61.0	108.0
Per step	17.2	17.0	12.0	23.0

I.2 COT GENERATION PROMPT

For each query–target pair (q_i, c_i^+), we prompt a frontier MLLM to produce a multi-step reasoning trace that bridges the query to the target. The prompt enforces three properties: (i) reasoning grounded solely in the query; (ii) no reference to the ground-truth target; and (iii) concise, retrieval-oriented output limited to 3–6 steps.

Generation Prompt

You are generating training data for an image+text-to-image retrieval pipeline on the CIRR dataset.

You will be given: a reference image; a query string describing how to modify the reference image to reach the target (relative caption/instruction); (internally) the matched target image, provided ONLY to ensure your reasoning is correct.

Critical constraints:

1. Do NOT explicitly mention, quote, paraphrase, or reveal any information derived from the matched target image or its existence. Never say things like "ground truth", "target", "annotation", or "according to the target". Your output must be valid even if the target were not available.

2. Write a retrieval-oriented intermediate reasoning trace: concise, information-dense steps describing what to extract from the reference image and how the text instruction changes it, so the retrieved image matches the intended target.

- Summarize key visual attributes of the reference image (main object/category, distinctive attributes, colors/materials, pose/viewpoint, scene/background, notable accessories, text/logos).
- Translate the instruction into explicit edit constraints: what must stay the same vs. what must change (add/remove/replace attributes, change color/viewpoint/context, change instance within same category, etc.).
- Prefer discriminative constraints that narrow down candidates and avoid generic cues unless necessary.
- End with a summary of the minimal properties the target image should satisfy relative to the reference.
- Be concise: remove redundant or low-information statements. Write 3-6 CoT lines only; if fewer suffice, output fewer.

3. Output ONLY the CoT steps as plain text. One sentence per line, separated by \n. No numbering, no bullet symbols, no extra commentary.

Input: reference_image: {{qry_image_path}} instruction: {{qry}} matched image: {{pos_image_path}}

I.3 CoT Filtering and Refinement Prompt

After generation, we apply a two-stage quality control pipeline. Rule-based filters first discard rationales with more than ten steps or any single step exceeding 300 tokens. A stronger MLLM then reviews each surviving rationale for validity and redundancy: (i) whether the CoT can plausibly guide retrieval of the correct target; (ii) whether steps exceeding five can be compressed; and (iii) whether invalid CoTs can be regenerated under the same no-leakage constraint.

Filtering & Refinement Prompt

You are a dataset quality verifier. Judge whether the provided CoT steps could reasonably lead from the query text and query image to retrieving the correct target image.

Inputs: Query instruction: {{qry}}; Query image: {{qry_image_path}}; Hidden ground-truth target image (INTERNAL ONLY): {{pos_image_path}}; Existing CoT steps: {{steps}}; Step length: {{step_length}}.

Use the hidden target image only to verify alignment. Never reveal it. Steps in {{steps}} are separated by ">>>>>".

1. Validity. Decide whether a reasonable solver could use the query instruction, query image, and the CoT steps to identify the correct target. Set "valid": true/false. Mark invalid if: major reasoning flaws; relies on unsupported information; cannot distinguish the correct target; or contains excessive filler ("let me think", "but wait", "hmm", "maybe", etc.).

2. Redundancy (only if valid = true and step_length > 5). Set "redundant": true if the CoT contains repetition or overly fine-grained steps that can be merged. If true, provide "replacement_steps" (at most 5 steps, joined by ">>>>>"). If false, set "replacement_steps": null.

3. Replacement for invalid CoT (only if valid = false). Provide "replacement_steps" (at most 5 steps) that: could plausibly guide retrieval; uses only query-inferable information; does not reveal the target; avoids filler; joined by ">>>>>". If impossible, set null.

Output: Exactly one JSON object: {"valid": bool, "redundant": bool|null, "replacement_steps": str|null}.

Rules: If valid=false, redundant=null. If valid=true and step_length$\leq$5, redundant=null and replacement_steps=null. Never output the target image path, filename, or any unique identifier. Output JSON only.

J Qualitative Comparison of Reasoning Approaches

We compare the inference-time outputs of RGE (Liu et al., 2025b) (explicit CoT), UME-R1 (Lan et al., 2026) (think-then-answer), and ALICE (latent reasoning) across all four MMEB meta-tasks, with two examples per task type. RGE and UME-R1 produce explicit text at query time; ALICE generates none. For ALICE, we visualize the top-8 vocabulary projections of each `<latent>` hidden state (rows L3–L15 correspond to 5 latent groups; values are softmax probabilities).

The examples below illustrate failure modes of explicit reasoning: RGE produces confident but factually incorrect descriptions, while UME-R1 generates verbose, repetitive deliberation that struggles to converge on a precise answer. Such errors and distractions can misguide the final embedding and impair retrieval. ALICE pursues latent reasoning without generating intermediate text, and its vocabulary projections in these examples highlight retrieval-relevant concepts.

Classification

ImageNet-1K *Classification*



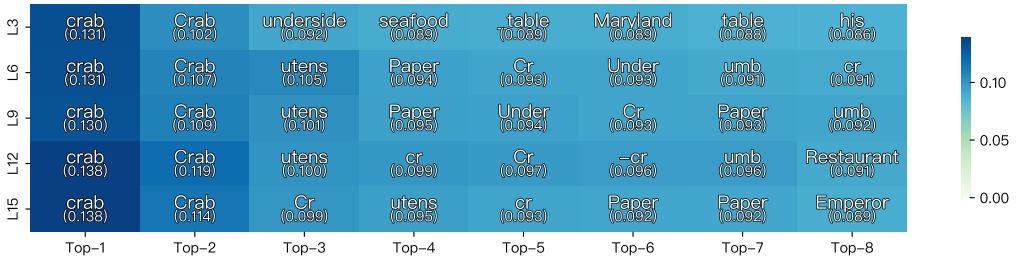
Query: Represent the given image for classification.

GT Target: rock crab, Cancer irroratus

▶ **RGE-3B:** The image shows a person preparing food at a table with a large crab on a plate. The crab's size and shape, along with its distinct coloration, align with characteristics of a rock crab. The setting suggests a casual dining environment, typical for such seafood preparation. <emb>

▶ **UME-R1-2B:** <think> So, let's analyze the input. The image shows a man in a striped shirt preparing food, probably at an outdoor or tented setting. There's food on a table, a cup, a container, and the background has structures like a tent or canopy. The key elements are a person (man) preparing food, a table with food, kitchenware, and a casual dining environment. The task is to represent the image for classification, so need to capture the main action and setting. The essence is a man preparing food in a casual dining setup, maybe a market or outdoor venue. So a concise summary would be "man preparing food in outdoor setting" or similar. Wait, need to check the image details. The man is handling food, maybe a seafood dish? The setting has a tent-like structure with a window. So the key elements: person (man), food preparation, casual dining, outdoor (tent-like) environment. So the concise summary could be "man preparing food outdoors". </think> <answer> <|begin_of_box|>man preparing food outdoors <|end_of_box|> <gen_emb>

▶ **ALICE-3B (ours):**



N24News ***Classification***



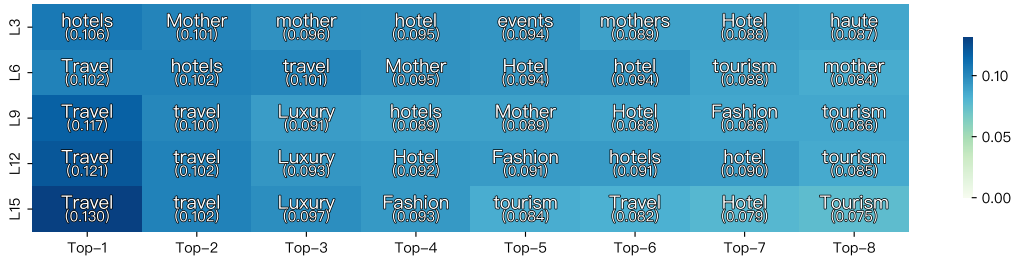
Query: Represent the given news image with the following caption for domain classification: The Grand Salon at the Baccarat Hotel New York will present fashion shows on Mother's Day as guests nibble on canapés, petit fours and chouquettes.

GT Target: Travel

► **RGE-3B:** The image depicts an elegant interior with luxurious decor, including chandeliers and sophisticated furniture, suggesting a high-end venue. The setting aligns with the text describing a fashion show at the Baccarat Hotel, indicating a focus on style and luxury, fitting the "Fashion & Style" category. <emb>

► **UME-R1-2B:** <think> So, let's analyze the input. The user's query mentions a news image related to the Grand Salon at the Baccarat Hotel New York, fashion shows on Mother's Day, guests with canapés, etc. The image shows a luxurious interior with chandeliers, floral arrangements, elegant seating—so the key elements are the hotel's salon, fashion show, Mother's Day event, and the setting's elegance. First, identify main topic: Baccarat Hotel New York's Grand Salon, fashion show on Mother's Day. Important entities: Baccarat Hotel New York, Grand Salon, fashion shows, Mother's Day, canapés, petit fours, chouquettes. Relationships: The salon is part of the hotel, hosting a fashion show with guests eating. Context: Luxury hotel interior, upscale event. Notable features: elegant decor (chandeliers, floral arrangements), formal seating. Synthesize into a concise summary. The essence is about the Baccarat Hotel's Grand Salon hosting a Mother's Day fashion show with canapés. So maybe "Baccarat Hotel New York Grand Salon Mother's Day fashion show". </think> <answer> Baccarat Hotel New York Grand Salon Mother's Day fashion show (A summary capturing the key elements: hotel, salon, Mother's Day event, fashion show, and the setting's elegance.) <gen_emb>

► **ALICE-3B (ours):**



Visual Question Answering

A-OKVQA *VQA*



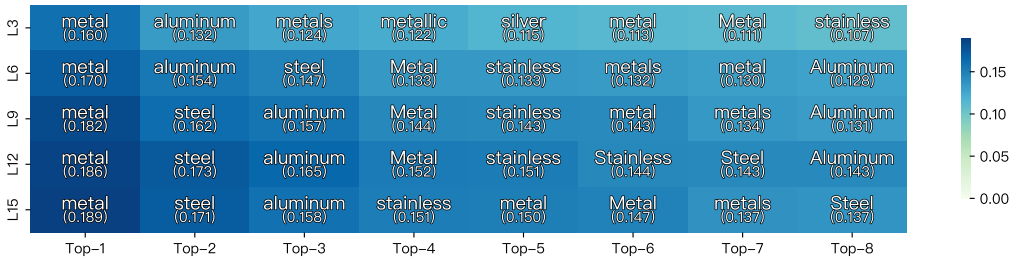
Query: Represent the given image with the following question: What is the bin on the left made from?

GT Target: steel

▶ **RGE-3B:** The image shows a barn with sheep. On the left, there is a metal container, which is typically made of metal. This aligns with the positive sample text, confirming the bin is made from metal, not wood or plastic, which are common materials for similar containers. <emb>

▶ **UME-R1-2B:** <think> So, let's analyze the input. The question is "What is the bin on the left made from?" First, look at the image. The bin on the left—yes, there's a container, maybe metal or plastic? Wait, the image has a metal container, maybe a barrel or similar. But the key is identifying the material. Wait, the image shows a container on the left, probably metal or plastic? Wait, no, maybe a barrel. But let's check. The bin on the left is a metal container, maybe a barrel. But the question is about the material. Wait, the image has a container that's likely metal. But let's confirm. The image shows a metal container, maybe a barrel. But the key is to identify the material. Wait, the bin on the left—looking at the image, the container is a metal barrel, maybe plastic, but more likely metal. Wait, but the question is asking for the material. Let's check the image again. The bin on the left is a metal container, maybe a barrel. But the key is to identify the material. Wait, the bin is a metal container, so maybe "metal" or "plastic" but need to check. Wait, the image has a container, maybe a barrel, which is typically metal. But let's confirm. The question is about what the bin on the left is made from. The answer would be metal, but let's make sure. Wait, the image shows a container on the left, which looks like a metal barrel. So the material is metal. But let's structure the analysis first. Key elements: The question is about the bin's material. The image has a container (bin) on the left. Need to identify the material. Looking at the image, the bin is a metal container, maybe a barrel. So the material is metal. Then, the synthesis would be the answer, which is "metal" or similar. Wait, let's check again. The image has a container on the left, which is a metal barrel. So the material is metal. </think> <answer> <|begin_of_box|>metal <|end_of_box|> <gen_emb>

▶ **ALICE-3B (ours):**



ChartQA ***VQA***



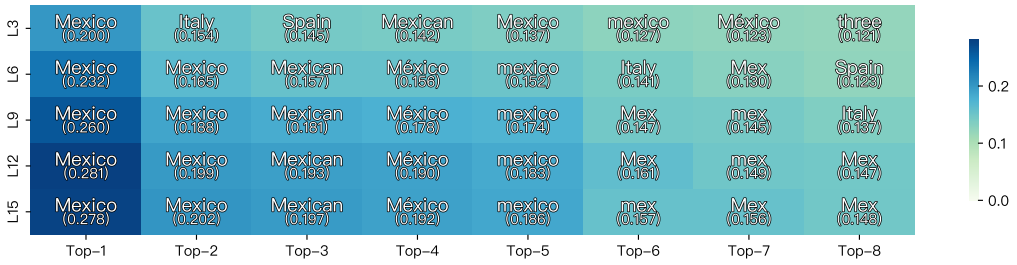
Query: Represent the given image with the following question: Which line remains all time lower than the other two?

GT Target: Mexico

► **RGE-3B:** The image shows a line graph comparing secondary graduation rates from 2010 to 2018. The red line (Italy) consistently remains below the purple (Spain) and blue (Mexico) lines, indicating it is the lowest throughout the period. This supports identifying the red line as the one that remains consistently lower than the other two. <emb>

► **UME-R1-2B:** <think> So, let's analyze the input. The question is asking which line (from the graph) remains lower than the other two. Looking at the graph, the lines are Italy (red dotted), Spain (purple solid), Mexico (blue solid). We need to check their values over time. Italy's line is consistently below Spain and Mexico's lines. So the key elements are the three lines (Italy, Spain, Mexico) and the comparison of their values. The main topic is identifying which line is lower. The essence is the comparison of the three lines' values over the time period. </think> <answer> <|begin_of_box|>Italy's line <|end_of_box|> <gen_emb>

► **ALICE-3B (ours):**



Retrieval

FashionIQ *Retrieval*



Query

Query: Find an image to match the fashion image and style note: Is shorter and has animal print and the dress has cheetah print and more revealing.

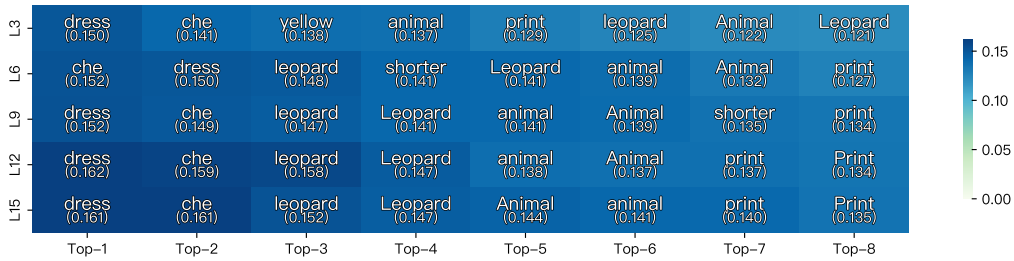


GT Target

▶ **RGE-3B:** The image shows a long, sleeveless dress in a bright yellow color. To find a similar style, look for a shorter version with cheetah print and more revealing details, maintaining the casual, summery vibe. <emb>

▶ **UME-R1-2B:** <think> Got it, let's analyze the input. The query is about finding an image that matches specific fashion elements: shorter length, animal print, and the dress has cheetah print and more revealing. So first, identify key elements: fashion item (dress), style note (shorter, animal print, cheetah print, more revealing). Then, need to capture the essence for retrieval. The main topic is fashion dress search with specific style. So the concise sentence would highlight the dress features. </think> <answer> Fashion dress retrieval with shorter length, animal print, cheetah print, and more revealing details. <gen_emb>

▶ **ALICE-3B (ours):**



CIRR *Retrieval*



Query

Query: Given an image, find a similar everyday image with the described changes: Two monkeys sit and watch two others play.



GT Target

▶ **RGE-3B:** The image shows a snowy landscape with two marmots. The marmots are lying on the snow, with one appearing to be resting and the other standing. The scene is set in a natural, wintry environment with patches of snow and rocks visible. The marmots have brown fur with lighter underbellies. To find a similar image with the described changes, look for a scene featuring two monkeys sitting and watching two others play. The setting should be a natural environment with trees and a grassy area, capturing a playful interaction between the monkeys. <emb>

▶ **UME-R1-2B:** <think> So, let's analyze the input. The query is asking for a similar everyday image with two monkeys sitting and watching two others play. First, identify key elements: main topic is monkeys, important entities are two monkeys (one sitting, two others playing), relationship is watching and playing, context is an everyday setting (since the original image has a snowy scene, maybe a natural environment), notable features are the action of sitting and watching. Then, need to synthesize into a concise capture. The essence is about monkeys in a social interaction scene. So the summary would highlight the key elements: monkeys, sitting, watching, playing, everyday setting. </think> <answer> Two monkeys sitting and watching others play (analysis: the input describes an image with monkeys in a social interaction context—two sitting, two playing; key elements are monkeys, their actions, and the social relationship; synthesis into a concise description of the essential scene elements) <gen_emb>

▶ **ALICE-3B (ours):**



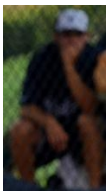
Visual Grounding

RefCOCO *Visual Grounding*



Query

Query: Select the portion of the image that follows the language expressions. “person hat right”

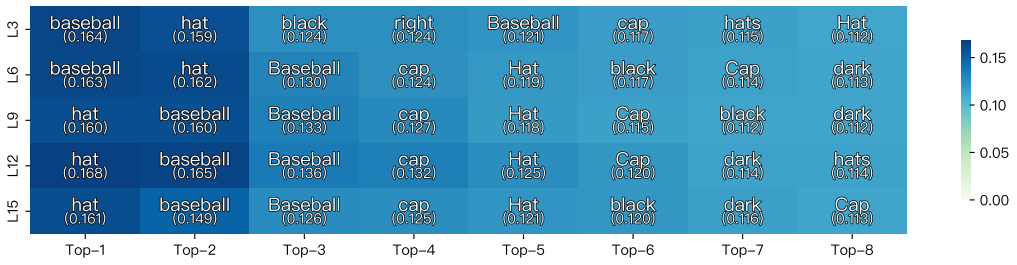


GT Target

► **RGE-3B:** The query text indicates the target is a “person wearing a hat on the right.” The image shows a baseball player in a white uniform with “HARRISON” and the number “26” on the back. Focus on the player’s head, which is wearing a black helmet, to locate the person wearing the hat on the right. <emb>

► **UME-R1-2B:** <think> So, let’s analyze the input. The user’s query is “Select the portion of the image that follows the language expressions. ‘person hat right’ ”. Wait, first, the image shows a baseball player with “HARRISON” and number 26, holding a bat. But the question is about “person hat right”—maybe the hat is part of the scene? Wait, the key elements here are the image (baseball player), the action (swinging bat), and the text “HARRISON” and number. But the input’s query is about selecting a portion following language expressions, maybe “person hat right” refers to a hat in the image. But the image has a player wearing a helmet, maybe the hat is part of that. But the main task is to identify key elements: the image contains a baseball player (HARRISON, 26) in a batting stance, with a helmet, bat, and background with other players. The query is about selecting a portion related to “person hat right”—so the hat might be part of the scene. But the essence for retrieval is the image with the baseball player, so the key elements are the player (HARRISON, 26), the action (swinging bat), and the hat. But the question is about the portion following language expressions, so maybe the hat is a key element. However, the summary should be concise. Let’s check: the input is a query about selecting a portion of the image related to “person hat right”—so the image has a player with a hat (helmet), so the portion with the hat. But the main topic is the image with the baseball player, so the essence is “baseball player with hat”. But let’s think again. The input is a query for retrieval, so the key elements are the image containing a baseball player (HARRISON, 26) in a batting stance, wearing a helmet with the number 26, and the background with other players. The query is about the portion following “person hat right”—so the hat (helmet) is a key element. But the summary should be a concise sentence or word. Since the image has a player with a helmet, the summary could be “baseball player with helmet” or similar. Wait, the image shows a player with a black helmet (baseball helmet) and the text “HARRISON” on the jersey. So the key elements are the baseball player, the helmet, and the action. So the essence is “baseball player with helmet”. </think> <answer> baseball player with helmet <gen_emb>

► **ALICE-3B (ours):**



Visual7W-Pointing *Visual Grounding*



Query

Query: Select the portion of the image that answers the question "Which white door is smallest?"

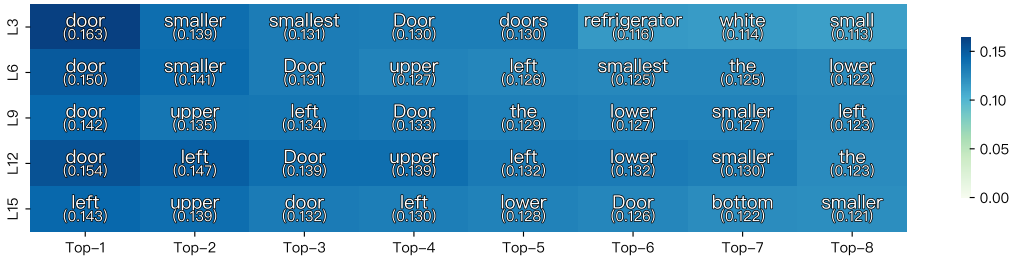


GT Target

▶ **RGE-3B:** The query text asks for the smallest white door. In the image, the white door is the refrigerator, which is the largest and most prominent. The smaller white door is the one on the left, partially visible, and is not the refrigerator. Focus on the left side of the image to locate the smallest white door. <emb>

▶ **UME-R1-2B:** <think> So, let's analyze the question: "Which white door is smallest?" First, identify the key elements: the question is about a white door (specifically, a door in the image) and determining which is smallest. Now, look at the image elements: there's a white door, likely the one on the left or right? Wait, the image shows a wooden wall with a white door. Let's check the doors. The door on the left side, maybe the one with the handle, is the smallest? Wait, need to check the image. The door on the left—probably the one with the handle, since it's a white door. Now, the question is asking which one is smallest. So the key is identifying the smallest white door. Then, the summary would need to capture the essence, but the input is a question, so the answer is about identifying the smallest white door. But since the task is to summarize the input for retrieval, the key is the question about the smallest white door. So the final result is identifying the smallest white door, which would be the one on the left (the one with the handle, maybe the leftmost white door, which is the smallest). Wait, let's check the image again. The door on the left, the one with the handle, is the smallest. So the essence is identifying the smallest white door. </think> <answer> <|begin_of_box|>The leftmost white door <|end_of_box|> <gen_emb>

▶ **ALICE-3B (ours):**



J.1 DETAILED PERFORMANCE ON MMEB

Table 10 presents our detailed results on the MMEB benchmark (Jiang et al., 2025), which consists of 36 tasks spanning four categories: classification, VQA, retrieval, and visual grounding. The benchmark covers a total of 20 in-domain and 16 out-of-domain datasets. ALICE-2B, ALICE-4B, and ALICE-8B use Qwen3-VL backbones, while ALICE-3B and ALICE-7B use Qwen2.5-VL backbones.

Table 10: **Detailed performance of the baseline models and ALICE on the 36 image tasks of MMEB.** We highlight those out-of-domain datasets with a blue background in the table.

Dataset	VLM2Vec-V2 -2B	VLM2Vec -7B	RGE -3B	CAFe -7B	UME-R1 -2B	UME-R1 -7B	Embed-RL -2B	Embed-RL -4B	PLUME -2B	ALICE -2B	ALICE -3B	ALICE -4B	ALICE -7B	ALICE -8B
Classification (10 tasks)														
ImageNet-1K	80.8	80.1	80.9	77.3	75.3	80.4	78.0	79.5	74.1	83.4	82.8	84.8	83.4	83.1
N24News	72.9	79.7	80.5	83.2	81.1	82.3	44.9	48.3	81.1	79.6	81.2	80.4	82.3	82.2
HatefulMemes	56.3	69.7	76.4	78.7	75.2	79.0	65.0	66.2	75.5	70.0	75.9	74.2	79.8	78.5
VOC2007	85.0	80.7	87.8	89.8	80.0	90.8	78.7	79.5	86.1	91.5	90.0	92.3	89.8	93.0
SUN397	71.0	77.4	76.9	79.9	79.4	80.3	75.4	79.2	76.9	80.6	79.8	82.8	82.3	82.6
Place365	35.9	37.4	45.2	45.0	42.6	46.8	43.9	43.1	42.4	47.5	46.3	46.0	44.8	47.9
ImageNet-A	47.4	58.1	35.9	55.2	50.4	53.9	59.2	58.1	50.8	52.8	41.7	51.5	48.0	53.8
ImageNet-R	89.3	73.9	79.0	88.0	88.7	90.1	88.5	88.2	87.5	89.1	82.2	90.3	80.6	87.7
ObjectNet	65.2	40.1	59.6	22.5	52.0	42.3	74.8	75.4	61.5	74.1	62.6	74.4	65.5	74.3
Country211	25.2	29.8	21.9	16.7	23.4	25.0	20.0	19.4	25.0	18.4	25.7	17.8	28.4	20.8
All Classification	62.9	62.7	64.4	63.6	64.8	67.1	62.8	63.7	66.5	68.7	66.8	69.5	68.5	70.4
VQA (10 tasks)														
OK-VQA	51.5	56.8	69.2	67.3	62.4	71.7	61.4	67.3	60.5	67.4	69.9	70.8	74.1	73.5
A-OKVQA	43.6	47.3	57.3	63.8	51.1	58.7	54.7	59.3	49.9	57.7	61.0	60.7	64.2	65.0
DocVQA	90.1	89.7	93.4	79.2	92.2	93.8	92.4	94.3	89.9	91.9	93.9	94.3	94.5	94.2
InfographicsVQA	58.8	60.0	73.9	53.3	67.7	79.2	76.7	77.5	59.6	67.3	75.7	72.0	78.1	75.4
ChartQA	47.4	56.9	75.4	48.8	64.9	75.1	80.7	80.9	49.8	63.1	69.8	68.2	75.3	72.8
Visual7W	52.9	52.7	55.4	52.5	54.1	55.2	52.7	55.3	47.6	54.8	58.2	58.2	59.4	59.3
ScienceQA	38.2	38.5	52.4	65.4	42.7	53.7	57.3	61.6	42.9	45.8	50.7	54.5	54.0	55.1
VizWiz	43.3	39.9	47.6	43.8	46.8	51.6	54.5	56.2	46.5	48.1	50.2	48.9	51.4	51.0
GQA	64.9	55.1	74.1	65.7	67.3	69.3	64.9	68.5	69.1	70.3	75.8	75.6	77.9	74.2
TextVQA	72.2	71.6	79.0	76.8	78.6	83.5	83.8	84.3	78.9	80.1	81.9	82.8	83.0	84.8
All VQA	56.3	56.9	67.8	61.7	62.8	69.2	67.9	70.5	59.2	64.7	68.7	68.6	71.2	70.5
Retrieval (12 tasks)														
VisDial	82.7	81.9	82.7	82.7	76.6	80.7	81.5	84.9	72.6	84.5	84.5	88.6	86.6	87.5
CIRR	57.5	51.1	54.0	60.4	53.7	55.3	47.6	61.2	54.6	58.5	58.0	62.6	61.7	66.2
VisualNews_t2i	74.5	80.5	75.6	69.5	71.7	76.8	71.9	73.7	71.3	74.0	76.7	75.4	79.8	79.7
VisualNews_i2t	78.2	81.2	79.4	79.4	74.2	82.0	73.6	73.9	72.7	76.8	81.6	81.6	82.2	83.1
MSCOCO_t2i	75.3	77.2	74.6	75.4	75.1	78.3	79.4	78.9	74.1	79.0	79.1	82.8	79.5	80.2
MSCOCO_i2t	71.4	73.9	73.6	73.1	68.9	71.4	75.3	76.3	69.8	76.3	75.6	78.4	77.0	78.3
NIGHTS	68.6	67.6	65.3	66.7	67.2	68.1	66.3	66.4	68.0	66.4	67.8	68.1	67.9	68.7
WebQA	90.6	88.3	88.8	89.3	90.0	90.9	89.3	90.5	89.1	88.2	90.5	90.5	90.6	91.0
FashionIQ	19.5	17.1	22.2	39.0	17.1	23.4	24.0	31.9	20.3	27.2	24.9	31.3	27.5	30.2
Wiki-SS-NQ	66.9	62.3	65.7	61.2	62.0	72.5	68.9	69.6	68.6	63.3	67.3	65.0	64.8	69.1
OVEN	64.3	66.5	69.0	60.8	66.9	71.4	61.4	60.7	68.4	62.1	70.4	65.0	73.9	65.7
EDIS	84.1	85.7	91.3	71.3	88.0	92.0	84.5	87.4	81.8	89.3	92.7	89.2	92.8	94.5
All Retrieval	69.5	69.4	70.2	69.1	67.6	71.9	68.6	71.3	67.6	70.5	72.4	73.2	73.7	74.5
Visual Grounding (4 tasks)														
MSCOCO	67.1	75.7	84.1	84.7	69.5	72.7	92.9	93.6	66.9	92.1	87.8	91.2	88.4	91.4
RefCOCO	87.1	87.6	94.6	89.4	83.3	91.4	94.9	95.9	86.5	93.9	94.2	95.1	95.3	95.9
RefCOCO-Matching	85.8	84.6	92.8	83.0	84.4	91.1	85.8	88.0	88.4	90.1	93.6	93.3	93.6	93.9
Visual7W-Pointing	69.2	81.0	88.8	93.2	71.5	84.2	88.0	87.9	74.9	86.1	86.6	83.4	84.4	88.3
All Visual Grounding	77.3	82.2	90.1	87.6	77.2	84.9	90.4	91.4	79.7	90.6	90.6	90.8	90.4	92.4
Final Score (36 tasks)														
Avg. IND Scores	68.8	71.4	75.3	72.8	71.5	76.1	71.9	74.3	69.5	75.2	77.0	77.9	78.8	79.3
Avg. OOD Scores	59.9	58.1	63.7	61.1	60.4	65.1	65.9	67.3	62.1	64.9	65.4	66.5	66.6	68.0
Avg. Scores	64.9	65.5	70.1	67.6	66.6	71.3	69.2	71.2	66.3	70.6	71.9	72.8	73.4	74.2

K PER-DATASET RESULTS ON MMEB-V2

Table 11 reports results on all 78 MMEB-v2 datasets. In this table, ALICE-2B and ALICE-7B use Qwen2-VL, and ALICE-8B uses Qwen3-VL. For comparison, we report VLM2Vec (Jiang et al., 2025), VLM2Vec-V2 (Meng et al., 2026), CAFe (Yu et al., 2025), UME-R1 (Lan et al., 2026), and PLUME (He et al., 2026). VLM2Vec, VLM2Vec-V2, UME-R1, and PLUME use Qwen2-VL. CAFe uses LLaVA-OneVision-7B.

Table 11: **Per-dataset performance on MMEB-v2.** Numbers in parentheses indicate the number of datasets in each category.

	VLM2Vec-7B	VLM2Vec-V2-2B	CAFe-7B	UME-R1-2B	UME-R1-7B	PLUME-2B	ALICE-2B	ALICE-7B	ALICE-8B
Avg - All (78 tasks)	52.3	58.0	60.6	60.1	64.5	61.6	65.0	68.7	68.9
Avg - Image (36 tasks, Hit@1)	65.5	64.9	67.6	66.6	71.3	66.3	71.1	74.8	74.4
Avg - Video (18 tasks, Hit@1)	34.0	34.6	42.4	42.2	47.5	44.1	42.7	46.5	47.5
Avg - Visdoc (24 tasks, nDCG@5)	46.4	65.4	63.9	63.9	67.1	67.5	72.6	76.3	76.7
I-CLS (10)	62.7	62.9	63.6	64.8	67.1	66.5	69.2	71.8	69.8
I-QA (10)	56.9	56.3	61.7	62.8	69.2	59.2	65.3	70.3	71.5
I-RET (12)	69.4	69.5	69.1	67.6	71.9	67.6	71.9	75.8	74.6
I-VG (4)	82.2	77.3	87.6	77.2	84.9	79.7	88.0	90.6	92.3
V-CLS (5)	39.1	39.3	35.8	44.3	48.6	45.0	51.6	56.3	54.6
V-QA (5)	30.0	34.3	58.7	51.2	60.7	52.3	48.0	53.7	54.9
V-RET (5)	29.0	28.8	34.4	32.9	38.2	33.5	33.2	35.4	35.1
V-MR (3)	40.6	36.8	39.5	39.7	39.3	46.7	34.6	36.9	43.8
VD-Vidore-V1 (10)	56.9	75.5	70.7	72.4	75.7	72.1	78.1	81.9	82.4
VD-Vidore-V2 (4)	9.4	44.9	49.6	46.2	50.5	49.8	48.6	53.4	54.4
VD-VisRAG (6)	59.1	79.4	79.5	79.2	83.7	78.1	84.9	88.1	87.8
VD-OOD (4)	38.1	39.4	38.1	37.2	37.6	57.4	64.2	67.3	68.0
ImageNet-1K	80.1	80.8	77.3	75.3	80.4	74.1	83.5	84.7	82.9
N24News	79.7	72.9	83.2	81.1	82.3	81.1	81.2	83.7	81.6
HatefulMemes	69.7	56.3	78.7	75.2	79.0	75.5	77.0	82.0	79.0
VOC2007	80.7	85.0	89.8	80.0	90.8	86.1	91.4	91.0	92.7
SUN397	77.4	71.0	79.9	79.4	80.3	76.9	80.9	82.3	82.4
Place365	37.4	35.9	45.0	42.6	46.8	42.4	46.7	46.7	46.3
ImageNet-A	58.1	47.4	55.2	50.4	53.9	50.8	44.7	53.9	52.5
ImageNet-R	73.9	89.3	88.0	88.7	90.1	87.5	88.7	89.9	88.6
ObjectNet	40.1	65.2	22.5	52.0	42.3	61.5	72.1	72.6	72.3
Country211	29.8	25.2	16.7	23.4	25.0	25.0	26.2	31.0	19.7
OK-VQA	56.8	51.5	67.3	62.4	71.7	60.5	67.1	73.2	73.2
A-OKVQA	47.3	43.6	63.8	51.1	58.7	49.9	56.0	63.0	63.7
DocVQA	89.7	90.1	79.2	92.2	93.8	89.9	93.2	94.6	95.5
InfographicsVQA	60.0	58.8	53.3	67.7	79.2	59.6	66.4	76.5	80.5
ChartQA	56.9	47.4	48.8	64.9	75.1	49.8	58.5	70.9	71.8
Visual7W	52.7	52.9	52.5	54.1	55.2	47.6	55.3	59.2	59.3
ScienceQA	38.5	38.2	65.4	42.7	53.7	42.9	44.1	52.8	57.2
VizWiz	39.9	43.3	43.8	46.8	51.6	46.5	53.0	51.2	52.3
GQA	55.1	64.9	65.7	67.3	69.3	69.1	76.3	78.0	77.4
TextVQA	71.6	72.2	76.8	78.6	83.5	78.9	83.0	83.8	83.6
VisDial	81.9	82.7	82.7	76.6	80.7	72.6	83.3	87.0	88.8
CIRR	51.1	57.5	60.4	53.7	55.3	54.6	57.3	66.3	63.8
VisualNews_t2i	80.5	74.5	69.5	71.7	76.8	71.3	77.1	83.2	79.6
VisualNews_i2t	81.2	78.2	79.4	74.2	82.0	72.7	82.4	86.0	82.9
MSCOCO_t2i	77.2	75.3	75.4	75.1	78.3	74.1	78.5	80.9	79.6
MSCOCO_i2t	73.9	71.4	73.1	68.9	71.4	69.8	75.3	79.4	78.2
NIGHTS	67.6	68.6	66.7	67.2	68.1	68.0	64.4	67.9	69.0
WebQA	88.3	90.6	89.3	90.0	90.9	89.1	89.9	90.4	91.0
FashionIQ	17.1	19.5	39.0	17.1	23.4	20.3	21.5	31.4	32.1
Wiki-SS-NQ	62.3	66.9	61.2	62.0	72.5	68.6	68.7	71.4	73.7
OVEN	66.5	64.3	60.8	66.9	71.4	68.4	71.4	74.2	63.2
EDIS	85.7	84.1	71.3	88.0	92.0	81.8	92.8	91.6	93.5
MSCOCO	75.7	67.1	84.7	69.5	72.7	66.9	87.5	90.2	90.0
RefCOCO	87.6	87.1	89.4	83.3	91.4	86.5	92.5	94.5	95.8
RefCOCO-Matching	84.6	85.8	83.0	84.4	91.1	88.4	90.5	92.7	93.7
Visual7W-Pointing	81.0	69.2	93.2	71.5	84.2	74.9	81.7	85.0	89.7
K700	35.5	38.0	40.1	35.8	42.8	42.2	49.1	54.8	52.5
SmthSmthV2	32.1	42.8	35.8	44.1	50.4	44.8	49.5	55.2	55.7
HMDB51	42.2	40.9	46.9	54.4	58.3	51.2	55.8	60.3	56.0
UCF101	61.8	60.0	39.6	67.2	70.0	66.5	74.7	80.0	79.1
Breakfast	23.8	14.8	16.6	20.1	21.5	20.1	28.9	31.4	29.8
MVBench	28.5	33.7	48.9	49.9	58.2	47.4	50.1	51.7	52.9
Video-MME	27.8	30.7	46.0	41.7	47.3	40.0	43.5	46.0	51.1
NExTQA	20.3	20.9	62.4	59.9	69.6	57.3	52.6	57.4	55.6
EgoSchema	21.8	34.0	60.0	45.4	52.4	47.8	32.6	39.0	38.6
ActivityNetQA	51.4	52.3	76.0	57.8	76.0	69.2	61.1	74.4	76.2
DiDeMo	29.3	30.4	37.8	32.4	40.0	32.7	33.3	35.3	35.4
MSR-VTT	34.5	28.3	36.5	34.3	38.9	36.2	37.0	38.9	39.5
MSVD	46.7	48.1	56.4	55.4	60.8	56.1	56.0	58.4	58.2
VATEX	25.5	26.5	32.0	29.9	32.6	28.2	28.4	30.4	30.4
YouCook2	9.0	10.6	9.5	12.7	18.5	14.5	11.3	13.9	12.3
QVHighlight	57.7	49.4	58.4	57.5	54.9	57.1	45.1	49.0	65.6
Charades-STA	19.8	20.2	18.7	20.4	21.9	19.4	15.1	17.7	20.5
MomentSeeker	39.3	40.8	41.4	41.2	41.1	63.5	43.8	43.9	45.3
ViDoRe_arxivqa	60.2	80.6	73.3	73.9	73.6	72.6	87.2	91.2	90.0
ViDoRe_docvqa	34.7	44.9	38.3	37.9	41.1	36.2	44.5	49.3	50.2
ViDoRe_infovqa	70.4	83.7	80.6	76.2	80.8	79.0	87.1	90.0	90.1
ViDoRe_tabfquad	78.2	89.2	80.7	86.1	90.2	88.8	90.5	93.5	94.3
ViDoRe_tatdqa	27.6	43.8	37.8	40.6	46.7	36.6	54.3	57.9	58.4
ViDoRe_shiftproject	38.6	60.8	52.0	66.8	65.0	64.8	63.3	70.6	72.3
ViDoRe_artificial_intelligence	67.7	88.5	86.0	85.9	89.5	83.8	90.6	94.2	94.8
ViDoRe_energy	60.4	86.5	84.8	83.3	85.7	82.6	86.7	89.9	89.9
ViDoRe_government_reports	61.8	85.0	85.0	82.6	89.8	83.2	86.2	87.9	90.0
ViDoRe_healthcare_industry	69.9	92.2	88.4	90.8	94.3	91.1	90.9	94.9	94.1
ViDoRe_esg_reports_human_labeled_v2	6.8	45.6	50.7	50.2	50.4	52.1	53.3	63.2	57.7
ViDoRe_biomedical_lectures_v2_multilingual	5.1	44.3	50.9	46.2	50.7	48.2	47.5	54.9	56.6
ViDoRe_economics_reports_v2_multilingual	13.9	43.0	54.3	45.7	57.8	49.6	49.5	54.9	57.9
ViDoRe_esg_reports_v2_multilingual	11.9	46.6	42.3	42.7	43.2	49.0	44.1	40.7	45.3
VisRAG_ArxivQA	52.6	76.9	74.0	74.3	80.5	71.6	83.2	87.9	87.2
VisRAG_ChartQA	70.2	84.4	82.7	86.0	85.0	80.8	87.6	89.7	90.0
VisRAG_MP-DocVQA	52.8	71.8	75.1	75.6	83.4	74.9	83.9	87.6	87.7
VisRAG_SlideVQA	72.8	91.5	87.6	87.1	91.5	88.9	93.2	95.1	95.5
VisRAG_InfoVQA	72.0	85.7	87.9	84.4	89.2	85.7	89.3	93.3	92.6
VisRAG_PlotQA	34.4	66.1	69.4	68.0	72.7	66.2	72.0	74.9	73.8
ViDoSeek-page	22.3	21.9	22.5	21.2	21.3	80.6	76.5	80.6	81.3
ViDoSeek-doc	77.8	80.2	73.8	75.9	75.3	76.9	80.9	83.4	83.2
MMLongBench-page	11.8	11.9	13.3	11.9	12.3	39.9	49.9	53.2	54.6
MMLongBench-doc	40.5	43.7	42.6	39.7	41.3	32.0	49.6	51.8	52.8