PACE: Pareto-Adaptive Compression for Efficient Native MLLMs

Lei Tian[1]* **Chunzheng Zhu**[2]*† **Kai WU**[1]
YiFan Zhang[3] **Haihua Yang**[1]‡

[1]ByteDance
[2]Hunan University
[3]Institute of Automation, Chinese Academy of Sciences

Abstract

Native multimodal large language models (native MLLMs) jointly encode visual and linguistic sequences within a unified autoregressive Transformer, emerging as a promising paradigm for multimodal understanding. Unlike conventional ViT-based MLLMs that rely on an external vision encoder, ViT-free native MLLMs tokenize every pixel patch directly, which is intuitively expected to suffer from severe visual token redundancy. A pilot study confirms this: unlike ViT-based models that degrade under compression, ViT-free MLLMs tolerate aggressive compression with near-lossless or even improved performance. Yet existing compression methods target only the ViT-based setting, leaving native MLLMs, which stand to gain the most, entirely unexplored. To close this gap, we introduce an analysis pipeline for native backbones centered on two metrics, *i.e.*, token discriminability and cross-dimensional heterogeneous redundancy, revealing that *(i)* redundancy is distributed independently and heterogeneously along the height and width axes, and *(ii)* early-layer visual tokens are indiscriminable and uniformly redundant. Guided by these findings, we propose **PACE** (Pareto Adaptive Compression for Efficient native MLLM), the first visual token compression framework tailored for native MLLMs with Multimodal-RoPE. Leveraging the dimensional decoupling of position encoding, PACE decomposes attention into two orthogonal spatial saliency estimates along height and width, gated by a positional attention signal, yielding a direction-aware importance measure with negligible overhead. It further casts token budget allocation as a Pareto optimization over performance preservation, inference efficiency, and spatial coverage, jointly identifying the optimal compression depths and retention ratios. Experiments show that PACE maintains near-lossless performance at only 25%–50% visual token retention, and even improves on several benchmarks, while delivering over 2× FLOPs reduction.

1 Introduction

Native multimodal large language models (MLLMs) such as Emu3 [95], Chameleon [90], SOLO [18], SAIL [48] and NEO [28] jointly model pixel and token sequences within a unified autoregressive Transformer. Unlike ViT-based MLLMs [60; 58; 20; 119; 91; 94] that rely on an external vision encoder, they tokenize every pixel patch directly, and intuitively suffer from severe visual token redundancy that degrades performance and bottlenecks inference efficiency. A pilot study on MMBench-DEV-EN [61] (Figure 1(a)) comparing Qwen3-VL [6] against a comparable native MLLM built in this work (§2) confirms this: unlike ViT-based models that degrade under compression, native MLLMs

*Equal contribution. ‡Corresponding author.
†Work done during an internship at ByteDance.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

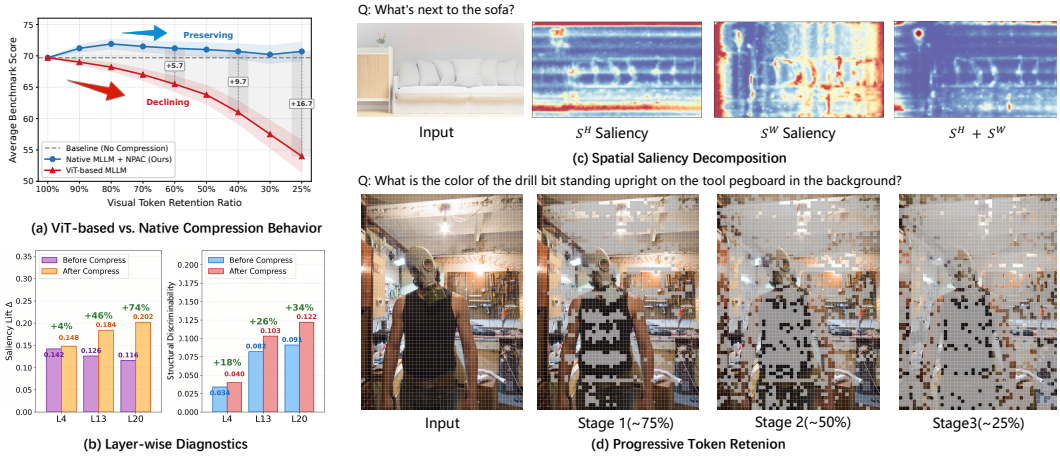


Figure 1: **Probing analysis and progressive compression in native MLLMs.** (a) ViT-based MLLMs degrade consistently under compression, while native MLLMs could preserve or even improve performance. (b) Layer-wise discriminability reveals a depth-progressive ramp after the IFB, guiding compression stage placement. (c) Spatially decoupled saliency along height and width dimensions. (d) Progressive token retention across multiple compression stages.

tolerate aggressive compression near-losslessly or even improve. Yet existing token-compression techniques, from attention-guided dropping [15; 104; 99] and token merging [82; 53; 107] to diversity-aware selection [117; 2; 108; 121], universally presuppose a modular ViT-based architecture, leaving native MLLMs, which should benefit most, entirely unexplored.

To solve this gap, we first conduct a systematic *probing analysis* of token redundancy in representative native backbones, centered on two quantitative metrics. *Discriminability* is assessed via two probes at each compression stage: (a) *Structural Discriminability*, the centered effective rank of the visual token matrix divided by the token count, where higher values indicate tokens span more independent directions with less redundancy; and (b) *Saliency Lift*, the relative gain in per-token 3D decomposed attention importance (computed along temporal, height, and width axes and then fused) from the full set to the PACE-selected subset, where higher values indicate the subset concentrates substantially more attention contribution than the layer-wise average. *Cross-dimensional heterogeneous redundancy* measures how unevenly redundancy is distributed along spatial height and width axes. Based on the analysis of these metrics (Figure 1(b) and (c)), we reveal two key findings.

Finding 1: Initial fusion buffer and depth-progressive redundancy accumulation. The initial layers serve as an *Initial Fusion Buffer (IFB)* [28], a concept established in prior native-MLLM analysis; our contribution is to identify its implication for token compression rather than to establish a new mechanism. During the IFB, visual and language tokens undergo semantic alignment through shared attention, so a stable discriminative structure has not yet emerged and Structural Discriminability in Figure 1(b) stays low regardless of compression. Its monotonic rise with depth, mirrored by the growth of Saliency Lift Δ on the right of Figure 1(b), forms a depth-progressive ramp that justifies conservative pruning near the IFB and aggressive compression on the deeper high-discriminability plateau, motivating the multi-stage schedule in PACE.

Finding 2: Heterogeneous redundancy across spatial dimensions. We observe empirically that redundancy is distributed independently and heterogeneously along the height and width dimensions: certain tokens are redundant along one axis yet critical along the orthogonal one. For instance, horizontal text lines in documents are salient along height but uniform along width, while vertical structures show the opposite. A single attention score conflating all dimensions cannot distinguish these modes [94; 121]. We treat this as empirical motivation for our spatially decoupled importance estimation, which separately evaluates height and width saliency, rather than as a broadly established property (visualization in Figure 1(c)).

Motivated by these findings, we propose **PACE** (Pareto-Adaptive Compression for Efficient Native MLLMs), the first visual token compression framework for native MLLMs with multi-axis positional encoding, comprising three components: (1) **Spatially Decoupled Importance Estimation (SDIE)**

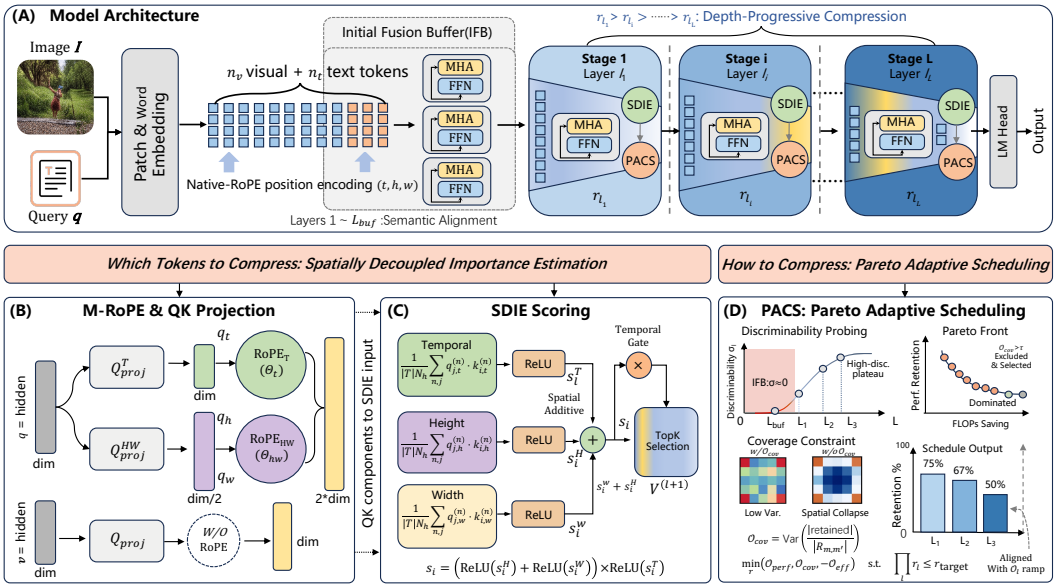


Figure 2: **Overview of PACE. (A)** Model architecture with depth-progressive compression ($r_{l_1} > r_{l_2} > \cdots > r_{l_L}$) applied after the IFB. **(B)** Native-RoPE QK projection: queries / keys are split into temporal ($\boldsymbol{q}_t$) and spatial ($\boldsymbol{q}_h, \boldsymbol{q}_w$) components with independent rotary encodings. **(C)** SDIE scoring: directional saliency scores are fused via spatial addition and temporal multiplicative gating for TopK selection. **(D)** PACS: discriminability probing identifies compression layers; a Pareto front over performance, efficiency, and coverage yields the optimal retention schedule.

(§2.2) exploits the dimensional decoupling of Native-RoPE to split attention into height and width saliency gated by a positional signal, giving a direction-aware, parameter-free importance measure. (2) **Pareto Adaptive Compression Scheduling (PACS)** (§2.3) uses a Pareto front search to locate compression layers along the discriminability curve and to allocate the cross-layer budget over performance, efficiency, and spatial coverage, yielding the per-stage retention ratio. (3) **Four-stage end-to-end training** (§2.4) builds a full PT/MT/SFT/RL pipeline around the compressed backbone to study how compression interacts with the native architecture, with compression applied directly at inference. As Figure 1(d) illustrates, progressive compression concentrates retained tokens on semantically critical regions while preserving spatial coverage. Experiments (§3) show that native MLLMs tolerate far more redundancy than ViT-based ones: on our full-recipe Ours-2B backbone PACE improves performance at 25%–50% token retention, and near-losslessness generalizes to other native backbones, all with about 2× FLOPs reduction.

2 Method

2.1 Framework Overview

As illustrated in Figure 2, PACE operates on a native MLLM backbone and performs progressive visual token compression along the model depth. Given an input image $\boldsymbol{I}$ and a text query $\boldsymbol{q}$, the native backbone first encodes them into visual tokens $\mathcal{V}^{(0)} = \{v_i\}_{i=1}^{n_v}$, where each $v_i \in \mathbb{R}^d$, and text tokens $\mathcal{T} = \{t_j\}_{j=1}^{n_t}$, where each $t_j \in \mathbb{R}^d$, via the Patch and Word Embedding layer, respectively. The mixed sequence then passes through the first L_{buf} layers, which constitute the *Initial Fusion Buffer (IFB)* for cross-modal semantic alignment and position registration, where L_{buf} is identified from the probing analysis as the layer beyond which discriminability begins to rise steadily (details in §3.1). After the IFB, compression is executed at a predefined set of stages $\mathcal{S} = \{l_1, l_2, l_3\}$ with $L_{\text{buf}} < l_1 < l_2 < l_3 \leq L$:

$$\mathcal{V}^{(l+1)} = \text{TopK}\big(\mathcal{V}^{(l)},\ s^{(l)},\ K^{(l)}\big), \quad l \in \mathcal{S}, \tag{1}$$

where TopK selects the $K^{(l)}$ tokens with the highest importance scores, $\mathcal{V}^{(l)}$ denotes the visual token set at layer l, $s^{(l)} = s_i i \in \mathcal{V}^{(l)}$ is the spatially decoupled importance score computed by SDIE

(§2.2), and $K^{(l)} = r_l \cdot |\mathcal{V}^{(l)}|$ is the number of tokens to retain at that stage, with the retention ratio r_l determined adaptively by PACS (§2.3). The progressive compression structure is deliberate: **compression intensity deepens gradually layer by layer** ($rl_1 > r_{l_2} > r_{l_3}$), aligned with the depth-progressive discriminability ramp, where early post-IFB layers warrant conservative pruning and deeper layers on the high-discriminability plateau admit more aggressive compression.

2.2 Spatially Decoupled Importance Estimation

Our probing analysis reveals that redundancy distributes independently and heterogeneously along the height and width dimensions: a token may be critical along only one spatial axis yet redundant along the other. Native-RoPE provides a natural structural prior for this decoupling. In native attention, the Query and Key vectors are partitioned along the head dimension into three components encoding temporal (t), height (h), and width (w):

$$\boldsymbol{q}_i = [\boldsymbol{q}_{i,t};\ \boldsymbol{q}_{i,h};\ \boldsymbol{q}_{i,w}] \in \mathbb{R}^{d_h}, \quad \boldsymbol{k}_j = [\boldsymbol{k}_{j,t};\ \boldsymbol{k}_{j,h};\ \boldsymbol{k}_{j,w}] \in \mathbb{R}^{d_h}, \tag{2}$$

where d_h denotes the total QK head dimension, and each component is equipped with its own rotary positional encoding $\Theta_T = \{\beta_T^{-2k/d}\}_{k\in[0,d/2)}$, $\Theta_H = \{\beta_H^{-4i/d}\}_{i\in[0,d/4)}$, $\Theta_W = \{\beta_W^{-4j/d}\}_{j\in[0,d/4)}$ with distinct base frequencies $\beta_T, \beta_H, \beta_W$. The attention score naturally decomposes into three additive terms:

$$\mathrm{attn}(i,j) = \underbrace{\boldsymbol{q}_{i,t}\cdot\boldsymbol{k}_{j,t}^\top}_{\text{temporal term}} + \underbrace{\boldsymbol{q}_{i,h}\cdot\boldsymbol{k}_{j,h}^\top}_{\text{height term}} + \underbrace{\boldsymbol{q}_{i,w}\cdot\boldsymbol{k}_{j,w}^\top}_{\text{width term}}. \tag{3}$$

We aggregate text-side attention from each component across all attention heads to compute directional importance scores for each visual token $i \in \mathcal{V}^{(l)}$:

$$s_i^T = \frac{1}{|\mathcal{T}|\cdot N_h}\sum_{n=1}^{N_h}\sum_{j\in\mathcal{T}} \boldsymbol{q}_{j,t}^{(n)}\boldsymbol{k}_{i,t}^{(n)\top}, s_i^H = \frac{1}{|\mathcal{T}|\cdot N_h}\sum_{n=1}^{N_h}\sum_{j\in\mathcal{T}} \boldsymbol{q}_{j,h}^{(n)}\boldsymbol{k}_{i,h}^{(n)\top}, s_i^W = \frac{1}{|\mathcal{T}|\cdot N_h}\sum_{n=1}^{N_h}\sum_{j\in\mathcal{T}} \boldsymbol{q}_{j,w}^{(n)}\boldsymbol{k}_{i,w}^{(n)\top} \tag{4}$$

where N_h is the number of attention heads, the superscript (n) indexes the n-th head, and $s_i^T, s_i^H, s_i^W \in \mathbb{R}$ denote the temporal, height directional, and width directional saliency of visual token i, respectively. Since these scores are raw dot products (not softmaxed) and may be negative, we apply ReLU to ensure non-negativity before fusion:

$$s_i = \big(\mathrm{ReLU}(s_i^H) + \mathrm{ReLU}(s_i^W)\big) \times \mathrm{ReLU}(s_i^T). \tag{5}$$

The additive spatial aggregation preserves tokens salient in *either* the H or W dimension, consistent with the cross-dimensional heterogeneity observed in the probing analysis, while the multiplicative temporal gate suppresses globally irrelevant tokens ($s_i^T \approx 0$). SDIE scores are extracted from intermediate QK products already computed in the native forward pass, introducing zero additional parameters. A formal optimality proof for this fusion form is given in Appendix B.1.

2.3 Pareto Adaptive Compression Scheduling

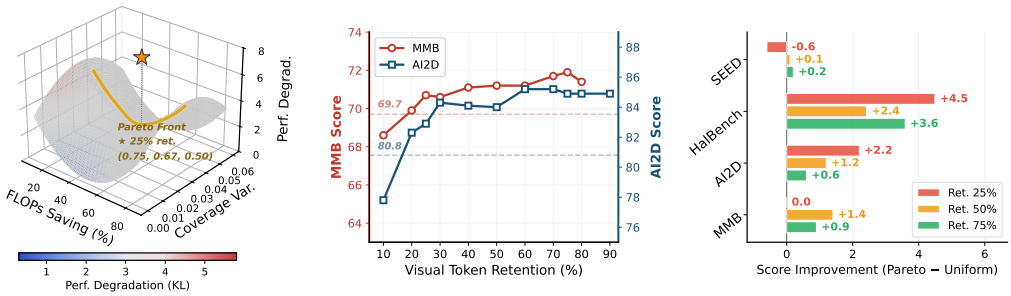
Since visual token redundancy accumulates progressively with depth and the optimal compression ratio varies across stages, we formulate cross-layer budget allocation to reach the Pareto frontier of performance, efficiency, and spatial coverage trade-offs.

Problem formulation. Let the compression stages be $\mathcal{S} = \{l_1, l_2, l_3\}$ with per-stage retention ratios $\boldsymbol{r} = (r_{l_1}, r_{l_2}, r_{l_3}) \in [0,1]^3$ as decision variables. The optimization problem is defined as:

$$\min_{\boldsymbol{r}} \big(\mathcal{O}_{\text{perf}}(\boldsymbol{r}),\ \mathcal{O}_{\text{cov}}(\boldsymbol{r}),\ -\mathcal{O}_{\text{eff}}(\boldsymbol{r})\big), \quad \text{s.t.} \prod_{l\in\mathcal{S}} r_l \le r_{\text{target}}, \tag{6}$$

where $\mathcal{O}_{\text{perf}}(\boldsymbol{r}) = \frac{1}{|\mathcal{B}|}\sum_{b\in\mathcal{B}}\big(S_b^{\text{full}} - S_b^{\text{comp}}(\boldsymbol{r})\big)$ denotes the average performance degradation across a benchmark suite $\mathcal{B}$ evaluated on a held-out calibration set, $\mathcal{O}_{\text{eff}}(\boldsymbol{r}) = 1 - \prod_{l\in\mathcal{S}} r_l$ is the total reduction ratio, $\mathcal{O}_{\text{cov}}$ is the spatial coverage variance (Eq. 8), and r_{target} is the target overall retention upper bound. All three objectives are min-max normalized to $[0,1]$ on the calibration set to ensure comparable scales during Pareto front construction. The Pareto optimal solution set is:

$$\mathcal{P}^* = \Big\{\boldsymbol{r} \;\Big|\; \prod_{l\in\mathcal{S}} r_l \le r_{\text{target}},\ \nexists\, \boldsymbol{r}' \text{ s.t. } \boldsymbol{r}' \prec \boldsymbol{r}\Big\}. \tag{7}$$



(a) Pareto Optimization Landscape **(b) Retention vs. Performance** **(c) Pareto vs. Uniform Gain**

Figure 3: **Pareto front visualization for compression schedule optimization.** **(a)** 3D Pareto front over performance preservation, FLOPs saving, and coverage variance; the gold star marks the selected operating point. **(b)** 2D projection of FLOPs saving vs. performance retention; color encodes coverage variance (green=low, red=high). **(c)** Per-layer optimal retention rate: three compression stages (orange/dark orange/red-orange) are aligned with the discriminability curve.

Spatial coverage constraint. Aggressive token dropping can concentrate retained tokens in salient regions while leaving large spatial gaps, degrading performance on tasks requiring global spatial understanding (*e.g.*, scene description, spatial grounding). To address this, the image is partitioned into $M_h \times M_w$ sub-regions $\{R_{m_h,m_w}\}$, where each $R_{m_h,m_w} \subset \mathcal{V}$ contains the visual tokens whose spatial coordinates (h_i, w_i) fall within the m_h-th row and m_w-th column of the grid. We set $M_h = M_w = 4$ to balance spatial granularity with the number of tokens per sub-region (sensitivity analysis in §3.1). Coverage variance is defined as:

$$\mathcal{O}_{\text{cov}}(\boldsymbol{r}) = \text{Var}_{(m_h,m_w)} \left(\frac{|\{i \in R_{m_h,m_w} : x_i = 1\}|}{|R_{m_h,m_w}|} \right), \tag{8}$$

where $x_i \in \{0, 1\}$ denotes the retention decision for token v_i. Candidate points with $\mathcal{O}_{\text{cov}}$ exceeding a threshold τ_{cov} are automatically excluded during traversal, making spatial coverage an **intrinsic constraint** of the Pareto selection rather than a post-hoc correction. Our experiments confirm that incorporating spatial coverage as an explicit objective yields measurable gains on tasks requiring holistic scene understanding, validating its inclusion in the multi objective formulation.

Front traversal and operating point selection. We enumerate $\boldsymbol{r} \in \mathcal{R}_\delta^{|\mathcal{S}|}$ on a discrete grid $\mathcal{R}_\delta = \{1.0,\ 1.0-\delta,\ \ldots,\ r_{\min}\}$, evaluate the three objectives on a calibration set, filter the Pareto set $\mathcal{P}^*$, and select the operating point subject to the r_{target} constraint. The discriminability curve from the probing analysis determines the front shape: stages aligned with the discriminability ramp correspond to moderate pruning, while those on the high plateau admit the most aggressive compression (specific configurations in §3.1). The resulting Pareto front is illustrated in Figure 3.

2.4 Training Strategy

Since no concurrent Native MLLM fully open-sources all training stages (PT / MT / SFT / RL), to further investigate the compression behavior of native MLLM across all training stages, we train a 2B-parameter native MLLM *from scratch*. Specifically, PACE follows a four-stage training pipeline: *i)* **Pre-training** (∼518M samples) learns basic visual concepts, and the Post-LLM backbone remains frozen. *ii)* **Mid-training** (∼89M samples) and *iii)* **SFT** (∼12.8M samples) strengthens vision-language understanding and alignment by using high-quality multi-source training data. All parameters are optimized end-to-end. *iv)* **RL** (∼12K samples) applies GRPO [83] with rule-based reward and category-proportional priority sampling for hard examples. We provide the simplified schematic illustration of data distribution in Figure 4 and more details in Appendix.

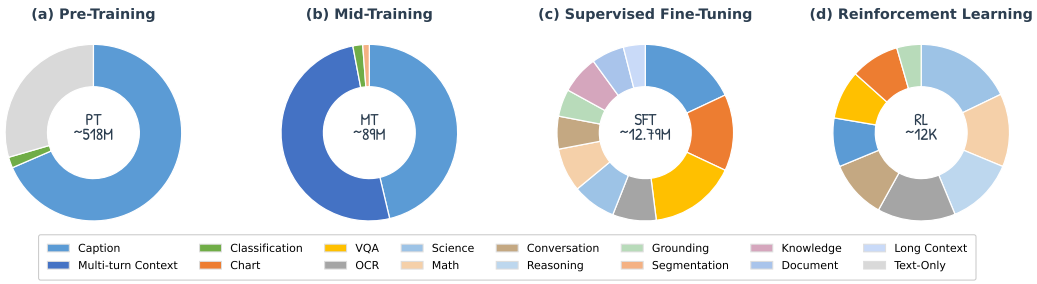


Figure 4: **Training data distribution across four stages.** Each donut chart shows the dataset composition and proportional contribution per stage.

Table 1: **Compression robustness across visual token retention ratios.** Each cell reports the relative change from the uncompressed baseline (absolute score in parentheses); **green**/red denote **improvement**/degradation. †ViT-free MLLMs. ⋆MME: sum of perception and reasoning. ‡HallusionBench: average of aAcc, qAcc, and fAcc as commonly done in prior works.

		MMB	AI2D	MME⋆	HallBench‡	RWQA	MMStar	SEED-IMG	MathVista
Retention=25%									
Ours-2B†	*w/o compress*	69.7	80.8	2084	42.7	61.8	52.9	71.8	55.7
	w/ compress	**+1.4%** (70.7)	**+2.8%** (83.1)	**+0.4%** (2092)	-2.5% (41.6)	-1.0% (61.2)	-4.7% (50.4)	-0.6% (71.4)	-9.7% (50.3)
SAIL-7B†	*w/o compress*	70.1	76.7	1719	33.1	63.9	53.1	72.9	57.0
	w/ compress	-2.6% (68.3)	-2.2% (75.0)	-2.5% (1676)	-1.0% (32.8)	-1.1% (63.2)	-13.6% (45.9)	-3.0% (70.7)	-11.2% (50.6)
ONECAT-3B†	*w/o compress*	83.1	77.2	2064	39.4	64.8	55.6	72.8	60.7
	w/ compress	-3.7% (80.0)	-5.5% (73.0)	-2.1% (2021)	-3.5% (38.0)	-3.1% (62.8)	-3.8% (53.5)	-4.1% (69.8)	-18.3% (49.6)
Qwen3-VL	*w/o compress*	78.4	76.9	2014.5	51.4	69.5	58.3	77.2	61.3
	w/ compress	-11.9% (69.1)	-8.7% (70.2)	-6.8% (1877)	-30.5% (35.7)	-8.1% (63.9)	-23.2% (44.8)	-4.7% (73.6)	-43.7% (34.5)
Intern3-VL	*w/o compress*	81.1	78.7	2221	42.5	66.3	60.7	76.5	57
	w/ compress	-6.3% (76.0)	-7.1% (73.1)	-5.1% (2107)	-4.9% (40.4)	-3.8% (63.8)	-11.5% (53.7)	-8.0% (70.4)	-26.5% (41.9)
Retention=50%									
Ours-2B†	*w/o compress*	69.7	80.8	2084	42.7	61.8	52.9	71.8	55.7
	w/ compress	**+2.2%** (71.2)	**+4.0%** (84.0)	**+0.6%** (2098)	**+0.5%** (42.9)	**+1.3%** (62.6)	-1.7% (52.0)	**+1.0%** (72.5)	-2.5% (54.3)
SAIL-7B†	*w/o compress*	70.1	76.7	1719	33.1	63.9	53.1	72.9	57.0
	w/ compress	-2.3% (68.5)	0.0% (76.7)	-0.4% (1712)	**+1.0%** (33.5)	-0.2% (63.8)	-10.4% (47.6)	-1.7% (71.7)	-5.3% (54.0)
ONECAT-3B†	*w/o compress*	83.1	77.2	2064	39.4	64.8	55.6	72.8	60.7
	w/ compress	-3.7% (80.0)	-2.6% (75.2)	-1.9% (2025)	-2.2% (38.5)	-2.8% (63.0)	**+0.9%** (56.1)	-2.1% (71.3)	-19.7% (48.8)
Qwen3-VL	*w/o compress*	78.4	76.9	2015	51.4	69.5	58.3	77.2	61.3
	w/ compress	-4.0% (75.3)	-7.7% (71.0)	-4.4% (1925)	-29.3% (36.3)	-4.9% (66.1)	-21.3% (45.9)	-4.3% (73.9)	-41.4% (35.9)
Intern3-VL	*w/o compress*	81.1	78.7	2221.2	42.5	66.3	60.7	76.5	57
	w/ compress	-6.3% (76.0)	-6.1% (73.9)	-6.0% (2087)	-2.6% (41.4)	-2.6% (64.6)	-11.9% (53.5)	-2.9% (74.3)	-23.9% (43.4)
Retention=75%									
Ours-2B†	*w/o compress*	69.7	80.8	2084	42.7	61.8	52.9	71.8	55.7
	w/ compress	**+3.2%** (71.9)	**+5.1%** (84.9)	**+1.7%** (2120.2)	**+3.9%** (44.4)	**+1.0%** (62.4)	-0.6% (52.6)	**+1.8%** (73.1)	**+1.3%** (56.4)
SAIL-7B†	*w/o compress*	70.1	76.7	1719	33.1	63.9	53.1	72.9	57.0
	w/ compress	-1.6% (69.0)	-0.7% (76.2)	-1.4% (1695)	**+1.6%** (33.7)	**+1.1%** (64.6)	0.0% (53.1)	-0.4% (72.6)	-1.7% (56.1)
ONECAT-3B†	*w/o compress*	83.1	77.2	2063.9	39.4	64.8	55.6	72.8	60.7
	w/ compress	-4.1% (80.6)	-2.0% (75.7)	-1.0% (2043)	-5.5% (37.2)	-0.3% (64.6)	0.0% (55.6)	-1.8% (71.5)	-26.0% (44.9)
Qwen3-VL	*w/o compress*	78.4	76.9	2015	51.4	69.5	58.3	77.2	61.3
	w/ compress	-4.7% (74.7)	-7.5% (71.1)	-3.6% (1942)	-26.2% (37.9)	-5.8% (65.5)	-21.3% (45.9)	-4.0% (74.1)	-35.1% (39.8)
Intern3-VL	*w/o compress*	81.1	78.7	2221	42.5	66.3	60.7	76.5	57.0
	w/ compress	-3.0% (78.7)	-6.1% (73.9)	-6.0% (2089)	-6.2% (39.8)	-3.3% (64.1)	-11.2% (53.9)	-1.5% (75.3)	-20.5% (45.3)

3 Experiments

3.1 Experimental Setup

Implementation details. PACE is built upon a ViT-free native multimodal large language model based on the Qwen3-1.7B [106] backbone. The depth of IFB is $L_{\text{buf}} = 4$. The native Native-RoPE base frequencies are $\beta_T = 1 \times 10^6$ and $\beta_H = \beta_W = 1 \times 10^4$. Full architecture configurations are in Table 11 (Appendix). All training stages use 5×8 H20-96G GPUs with bfloat16 precision. More training details are in Table 10 (Appendix).

Table 2: **Comparison with transferred modular training-free compression methods on native MLLM.** † denotes transfer from modular architectures. Best results per retention level are **bolded**.

Method	Ret.	MMB	AI2D	MME	HalBench	RWQA	MMStar	SEED	MathVista
Ours-2B (full)	100%	69.7	80.8	2084	42.7	61.8	52.9	71.8	55.7
25% Retention									
FastV†	25%	-27.7% (50.4)	-39.9% (48.6)	-31.3% (1432)	-27.9% (30.8)	-63.3% (22.7)	-26.1% (39.1)	-25.9% (53.2)	-32.7% (37.5)
PyramidDrop†	25%	-3.2% (67.5)	-2.1% (79.1)	-4.4% (1992)	-7.8% (39.4)	-2.9% (60.0)	-12.3% (46.4)	-5.7% (67.7)	**-13.5%** (48.2)
D^2Pruner†	25%	-3.2% (67.5)	-11.9% (71.2)	-3.5% (2011)	-13.2% (37.1)	-9.1% (56.2)	-15.9% (44.5)	-7.2% (66.6)	-18.1% (45.6)
SparseVLM†	25%	-1.1% (68.9)	**+3.7%** (83.8)	-1.2% (2059)	-4.5% (40.8)	-2.1% (60.5)	-9.8% (47.7)	-3.8% (69.1)	**-6.5%** (52.1)
Ours	25%	**+1.4%** (70.7)	+2.8% (83.1)	+0.4% (2092)	**-2.6%** (41.6)	**-1.0%** (61.2)	**-4.7%** (50.4)	**-0.6%** (71.4)	-9.7% (50.3)
50% Retention									
FastV†	50%	-7.3% (64.6)	-24.8% (60.8)	-12.6% (1821)	-10.6% (38.2)	-50.2% (30.8)	-12.9% (46.1)	-8.6% (65.6)	-16.2% (46.7)
PyramidDrop†	50%	+1.6% (70.8)	+1.5% (82)	+0.4% (2092)	-3.1% (41.4)	-2.4% (60.3)	-3.6% (51.0)	-0.4% (71.5)	-3.2% (53.9)
D^2Pruner†	50%	-1.1% (68.9)	+2.0% (82.4)	**+0.5%** (2096)	-2.6% (41.6)	-2.9% (60)	-5.3% (50.1)	-0.4% (71.5)	-6.6% (52.0)
SparseVLM†	50%	+1.6% (70.8)	**+5.2%** (85)	**-0.2%** (2081)	**+0.3%** (42.8)	**-4.6%** (61.8)	-3.4% (51.1)	**+0.1%** (71.9)	-1.1% (55.1)
Ours	50%	**+2.2%** (71.2)	+4.0% (84)	0.6% (2098)	**+0.4%** (42.9)	**+1.3%** (62.6)	**-1.7%** (52)	**+1.0%** (72.5)	**-2.5%** (54.3)
75% Retention									
FastV†	75%	-1.6% (68.6)	-15.7% (68.1)	-3.1% (2021)	-0.3% (42.6)	-43.4% (35.0)	-36.1% (33.8)	-0.8% (71.2)	-5.6% (52.6)
PyramidDrop†	75%	+1.9% (71)	+3.6% (83.7)	-0.3% (2077)	+0.4% (42.9)	-0.6% (61.4)	-0.4% (52.7)	+1.4% (72.8)	-1.6% (54.8)
D^2Pruner†	75%	+1.7% (70.9)	+3.8% (83.9)	+0.9% (2103)	+0.2% (42.8)	-1.8% (60.7)	0.0% (52.9)	+1.0% (72.5)	-3.8% (53.6)
SparseVLM†	75%	+1.3% (70.6)	**+5.3%** (85.1)	+0.9% (2104)	-0.5% (42.5)	+0.6% (62.2)	-1.0% (52.4)	+1.0% (72.5)	+0.7% (56.1)
Ours	75%	**+3.2%** (71.9)	+5.1% (84.9)	**+1.7%** (2120)	**+3.9%** (44.4)	**+1.0%** (62.4)	-0.6% (52.6)	**+1.8%** (73.1)	**+1.3%** (56.4)

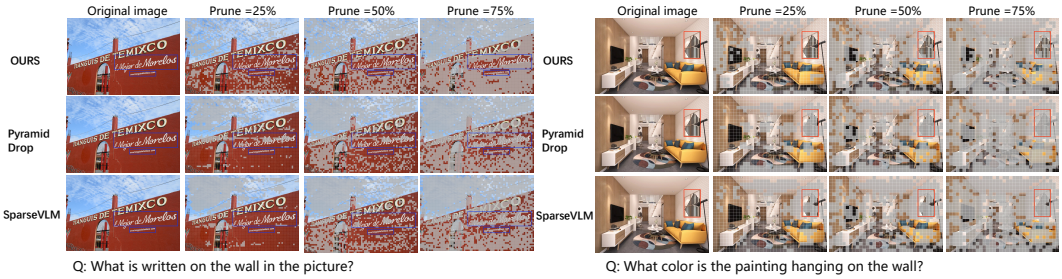


Figure 5: **Progressive visual token retention maps at three compression stages.** Retained tokens increasingly focus on semantically critical regions (e.g., text and objects) while preserving spatial coverage, validating the Pareto-adaptive schedule's balance between performance and coverage.

Evaluation methods. In Table 1, we apply PACE to four representative MLLMs spanning both paradigms, *i.e.*, the ViT-free native models SAIL-7B [48] and ONECAT-3B [52], and the ViT-based modular models Qwen3-VL [6] and InternVL3 [119]. In Table 2, we instead transfer existing modular training-free compression methods to our own native MLLM backbone. All comparison experiments are implemented based on the official checkpoints or codebases released by the official authors.

Front traversal and operating point selection. The search jointly optimizes layer positions and per-stage retention ratios. Three ordered stages $\mathcal{S} = \{l_1, l_2, l_3\}$ ($l_1 < l_2 < l_3$) are chosen from the 36 post-IFB layers, giving $\binom{36}{3} = 7{,}140$ configurations. Each layer-wise retention rate r_i is drawn from a shared 25-value set $\mathcal{G} = \{0.10, 0.15, \ldots, 0.90\} \cup \{k/9\}_{k=1}^{8}$, yielding $25^3 = 15{,}625$ ratio combinations and $\sim 1.12 \times 10^8$ total candidates. We evaluate the three objectives on a calibration set, extract $\mathcal{P}^*$ via dominance filtering, and pick the point satisfying r_{target}. Guided by **Finding 1** (depth-progressive redundancy accumulation), the monotonicity relation $r_1 > r_2 > r_3$ naturally emerges, substantially narrowing the traversal space. the resulting front is shown in Figure 3.

Evaluation benchmarks. We evaluate with VLMEvalKit [31] on eight benchmarks, *i.e.,* MMBench-DEV-EN (MMB) [61], AI2D [42], MME [32], HallusionBench (HallBench) [35], RealWorldQA (RWQA), MMStar [17], SEEDBench-IMG (SEED) [51], and MathVista [62].

3.2 Main Results

Table 1 compares PACE against modular MLLMs at 25%, 50%, and 75% retention, reporting relative change from each model's uncompressed baseline. PACE uniquely benefits from compression: it gains on MMB (+1.4%→+3.2%) and AI2D (+2.8%→+5.1%) as retention increases and stays near-lossless at 25% on most benchmarks except MMStar and MathVista, whereas modular MLLMs

degrade consistently, e.g. Qwen3-VL collapses up to 43.7% on MathVista at 25%, reflecting our core motivation that ViT-free MLLMs accumulate far more redundant tokens. Applying SDIE and PACS training-free to other ViT-free MLLMs corroborates this: SAIL-7B, sharing the decoupled spatial encoding, keeps a tolerable margin on general benchmarks (MMB, AI2D, MME, HallBench, RWQA, SEED), while ONECAT-3B, which folds spatial information into a single 1-D signal, drops more (AI2D -5.5% vs. -2.2% at 25%). The ordering is not uniform, however: on MMStar SAIL-7B drops more (-13.6% vs. -3.8%) and both degrade on MathVista (-11.2%, -18.3%), reflecting sensitivity to dense mathematical detail rather than a generic 1-D failure. This cross-benchmark inconsistency in spatial sensitivity motivates SDIE's directional decoupling.

3.3 Comparison with Transfer Baselines

Table 2 compares PACE with modular compression methods transferred to the native MLLM backbone across retention levels. FastV [15] degrades severely under aggressive compression, with MMB dropping from 69.7 to 50.4 (27.7PyramidDrop [104] addresses only the former via multi-stage compression and thus still trails PACE. D^{2}Pruner [113] uses diversity-aware scalar ranking, while SparseVLM [117] re-weights tokens by text-visual relevance; both overlook the decoupled structure. Consequently, PACE 's advantage is largest at 25% retention, where dimension-aware scoring becomes most critical. Figure 5 further shows that PyramidDrop† and SparseVLM† preserve broader spatial structure but leave noticeable gaps, whereas PACE focuses on semantic regions while maintaining spatial coverage.

3.4 Ablation Studies

Uniform vs. Pareto-adaptive compression. Table 3 compares uniform compression (equal retention across stages) against the PACS-optimized schedule. For the same overall retention target, Pareto scheduling consistently outperforms uniform allocation. At 25% overall retention, the final Pareto configuration $(0.75, 0.67, 0.50)$ outperforms uniform by 3.6% on MMB and 7.6% on AI2D. This validates that adaptive budget allocation guided by layer-wise discriminability is critical.

Table 3: **Uniform vs. Pareto-adaptive compression on Ours-2B.** "Pareto" uses the PACS-optimized schedule; "Uniform" applies equal retention at every stage.

Schedule	Ret.	MMB	AI2D	MME	HalBench	SEED	RWQA
Full tokens	100%	69.7	80.8	2084.4	42.72	71.8	61.8
Pareto	25%	+1.4% (70.7)	+2.9% (83.1)	+0.4% (2092)	-2.5% (41.6)	-0.6% (71.4)	-1.0% (61.2)
Uniform	25%	-2.2% (68.2)	-4.7% (77.0)	-5.2% (1976)	-14.8% (36.4)	-6.4% (67.2)	-8.4% (56.6)
Pareto	50%	+2.2% (71.2)	+4.0% (84.0)	+1.1% (2107)	+0.5% (42.9)	+1.0% (72.5)	+1.3% (62.6)
Uniform	50%	-0.3% (69.5)	+1.7% (82.2)	-1.6% (2052)	-6.5% (40.0)	-0.6% (71.4)	-1.8% (60.7)
Pareto	75%	+3.2% (71.9)	+5.1% (84.9)	+1.7% (2120)	+3.9% (44.4)	+1.8% (73.1)	+1.0% (62.4)
Uniform	75%	+1.9% (71.0)	+4.3% (84.3)	+0.1% (2087)	-4.5% (40.8)	+1.1% (72.6)	+0.5% (62.1)

SDIE internal ablation. Table 4 compares alternative importance formulations on the our backbone. The full configuration (Setting 1: $(s_i^H + s_i^W) \times s_i^T$ with all SDIE components) consistently outperforms alternatives. Setting 2 (only $s_i^T \times s_i^{H,W}$, without spatial axis decoupling. s_i^{HW} is a single un-decoupled spatial score, *not* $s_i^H \times s_i^W$) notably drops MMB by 2% - 4%, while Setting 4 (only $s_i^T \times s_i^W$, discarding s_i^H) causes a much sharper decline at 25% retention, demonstrating that both spatial dimensions are jointly essential.

Table 4: **SDIE internal ablation.** Setting 1 = full PACE; Setting 2 = $s_i^T \times s_i^{H,W}$; Setting 3 = $s_i^T \times s_i^H$ only; Setting 4 = $s_i^T \times s_i^W$ only.

Configuration	Ret.	MMB	AI2D	MME
Full tokens	100%	69.7	80.8	2084
Full PACE (Setting 1)	25%	+1.4% (70.7)	+2.9% (83.1)	+0.4% (2092)
	50%	+2.2% (71.2)	+4.0% (84.0)	+0.6% (2098)
	75%	+3.2% (71.9)	+5.1% (84.9)	+1.7% (2120)
$s_i^T \times s_i^{H,W}$ (Setting 2)	25%	-2.4% (68.0)	-2.5% (78.8)	-3.7% (2008)
	50%	-0.1% (69.6)	+3.0% (83.2)	-0.2% (2080)
	75%	+0.9% (70.3)	+5.1% (84.9)	+0.5% (2096)
$s_i^T \times s_i^H$ (Setting 3)	25%	-2.4% (68.0)	-3.5% (78.0)	-2.6% (2031
	50%	+1.2% (70.5)	+1.4% (81.9)	-0.7% (2069)
	75%	+0.7% (70.2)	+4.8% (84.7)	+0.5% (2095)
$s_i^T \times s_i^W$ (Setting 4)	25%	-4.6% (66.5)	-2.0% (79.2)	-2.7% (2029)
	50%	+0.6% (70.1)	+2.5% (82.8)	-1.2% (2059)
	75%	+0.0% (69.7)	+5.5% (85.2)	-0.4% (2075)

Number of compression stages. Figure 6 compares $N \in 1, 2, 3, 4$ stages at the same token retention. A single stage ($N = 1$) causes abrupt information loss (e.g., SEED drops 26.0% at 25% retention), while $N = 2$ remains insufficient under aggressive compression. A fourth stage ($N = 4$) brings negligible gains with additional scheduling complexity. Overall, setting $N = 3$ provides the best trade-off, matching or outperforming the baseline across benchmarks and retention levels, while avoiding unnecessary complexity from additional compression stages.

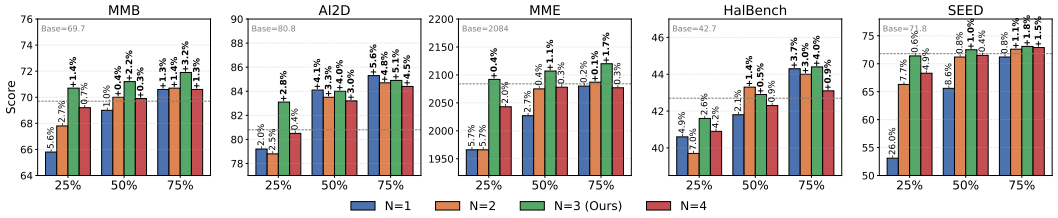


Figure 6: **Effect of the number of compression stages (N) on Ours-2B.** The three-stage design ($N=3$) consistently preserves or improves baseline performance across all retention ratios, while $N=1$ incurs notable degradation under aggressive compression.

Table 5: **Efficiency analysis of different compression methods on Ours 2B model.** Measured on a single H20-96G GPU, batch size 1, input resolution 1024^2. Attention-based methods shown with red background; spatially-decoupled methods with green background. Speedup ratios relative to the full-token baseline in parentheses. Avg. Score is averaged over MMB, AI2D, HallBench, RealWorld and SEEDBench-IMG five benchmarks.

Method	# Token	FLOPs (T)	Prefill (ms)	Decode (ms/tok)	KV Cache (MB)	GPU Mem (GB)	Avg. Score
Ours-2B (Full)	1024	5.43	127.8	77.4	251.7	7.1	65.4
Retention = 75% (∼768 tokens)							
SparseVLM	768	4.36 (×1.25)	103.1 (×1.24)	77.8 (×1.00)	191.7	6.7	66.6
PyramidDrop†	768	4.36 (×1.25)	106.1 (×1.20)	78.7 (×0.98)	191.7	6.7	66.4
PACE	**768**	**4.60 (×1.18)**	**105.6 (×1.20)**	**77.6 (×1.00)**	191.7	**6.8**	**67.3**
Retention = 50% (∼512 tokens)							
SparseVLM	512	3.56 (×1.53)	98.8 (×1.29)	75.9 (×1.02)	131.7	6.3	66.2
PyramidDrop†	512	3.56 (×1.53)	96.9 (×1.32)	73.4 (×1.05)	131.7	6.3	66.2
PACE	**512**	**3.61 (×1.50)**	**102.1 (×1.25)**	**73.6 (×1.05)**	131.7	**6.4**	**66.6**
Retention = 25% (∼256 tokens)							
SparseVLM	256	2.74 (×1.98)	98.2 (×1.30)	76.9 (×1.01)	71.7	6.0	64.6
PyramidDrop†	256	2.74 (×1.98)	98.2 (×1.30)	76.6 (×1.01)	71.7	6.0	62.7
PACE	**256**	**2.74 (×1.98)**	**99.6 (×1.28)**	**77.3 (×1.00)**	71.7	**6.2**	**65.6**

3.5 Efficiency Analysis

We evaluate inference efficiency on a single H20-96G GPU at 1024^2 resolution (=1024 visual tokens after 2×2 patch merge). Table 5 reports batch-size1 results; matched-retention batched serving is in AppendixF.5. At 25% retention ($\sim$256 tokens), PACE reduces FLOPs by $2.0\times$ and KV cache from 252,MB to 72,MB, while improving the average score from 65.4 to 65.6. Since batch-1 decode is dominated by weight loading, all methods have the same $1.00\times$ decode speed, and KV-cache reduction is generic to 256-token methods. PACE 's key advantage is accuracy at matched serving cost: 65.6 average, outperforming SparseVLM by 1.0 and PyramidDrop by 2.9, while exceeding the full model (65.4). Under batched decode at 25% retention, PACE achieves up to $1.9\times$ faster decoding and $2.0\times$ lower peak memory, and supports batch64 where the full model OOMs (AppendixF.5).

4 Conclusion

We have presented PACE, the first visual token compression framework designed for native MLLM built on multi-axis positional encoding. Through systematic probing analysis, we reveal the progressive redundancy accumulation pattern induced by the IFB mechanism and the cross-dimensional heterogeneous distribution of redundancy. Building on these findings, PACE leverages the inherent dimensional decoupling of Native-RoPE to construct a direction-aware importance measure and employs Pareto front traversal to determine the optimal compression ratio at each layer. Experiments across multiple benchmarks validate the effectiveness and efficiency of PACE, demonstrating that the structured priors inherent in native architectures yield systematic gains for efficient inference.

References

[1] Acharya, M., Kafle, K., Kanan, C.: Tallyqa: Answering complex counting questions. In: Proceedings of the AAAI conference on artificial intelligence. vol. 33, pp. 8076–8084 (2019)

[2] Alvar, S.R., Singh, G., Akbari, M., Zhang, Y.: Divprune: Diversity-based visual token pruning for large multimodal models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 9392–9401 (2025)

[3] An, X., Xie, Y., Yang, K., Zhang, W., Zhao, X., Cheng, Z., Wang, Y., Xu, S., Chen, C., Zhu, D., et al.: Llava-onevision-1.5: Fully open framework for democratized multimodal training. arXiv preprint arXiv:2509.23661 (2025)

[4] Awadalla, A., Xue, L., Lo, O., Shu, M., Lee, H., Guha, E., Jordan, M., Shen, S., Awadalla, M., Savarese, S., et al.: Mint-1t: Scaling open-source multimodal data by 10x: A multimodal dataset with one trillion tokens. Advances in Neural Information Processing Systems **37**, 36805–36828 (2024)

[5] Baek, C., Song, J., Kim, S., Kong, K.: Agilepruner: An empirical study of attention and diversity for adaptive visual token pruning in large vision-language models. arXiv preprint arXiv:2603.01236 (2026)

[6] Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., et al.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025)

[7] Bavishi, R., Elsen, E., Hawthorne, C., Nye, M., Odena, A., Somani, A., Tasırlar, S.: Fuyu-8b: A multimodal architecture for ai agents. URL: https://www. adept. ai/blog/fuyu-8b **3** (2023)

[8] Biten, A.F., Tito, R., Mafla, A., Gomez, L., Rusinol, M., Valveny, E., Jawahar, C., Karatzas, D.: Scene text visual question answering. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4291–4301 (2019)

[9] Borisova, E., Rauscher, N., Rehm, G.: Scivqa 2025: Overview of the first scientific visual question answering shared task. In: Proceedings of the Fifth Workshop on Scholarly Document Processing (SDP 2025). pp. 182–210 (2025)

[10] Byeon, M., Park, B., Kim, H., Lee, S., Baek, W., Kim, S.: Coyo-700m: Image-text pair dataset (2022)

[11] Chang, C., Fu, M., Chen, X., Feng, S., Zhang, M., Zhou, X., Chuang, J.J., Pan, J., Liu, Z.: Research on pdu-net lung nodule segmentation algorithm based on path aggregation and dual attention. In: 2025 4th International Conference on Image Processing, Computer Vision and Machine Learning (ICICML). pp. 1897–1900. IEEE (2025)

[12] Chen, G.H., Chen, S., Zhang, R., Chen, J., Wu, X., Zhang, Z., Chen, Z., Li, J., Wan, X., Wang, B.: Allava: Harnessing gpt4v-synthesized data for lite vision-language models. arXiv preprint arXiv:2402.11684 (2024)

[13] Chen, J., Tang, J., Qin, J., Liang, X., Liu, L., Xing, E., Lin, L.: Geoqa: A geometric question answering benchmark towards multimodal numerical reasoning. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. pp. 513–523 (2021)

[14] Chen, J., Xu, Z., Pan, X., Hu, Y., Qin, C., Goldstein, T., Huang, L., Zhou, T., Xie, S., Savarese, S., et al.: Blip3-o: A family of fully open unified multimodal models-architecture, training and dataset. arXiv preprint arXiv:2505.09568 (2025)

[15] Chen, L., Zhao, H., Liu, T., Bai, S., Lin, J., Zhou, C., Chang, B.: An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models. In: European Conference on Computer Vision. pp. 19–35. Springer (2024)

[16] Chen, L., Li, J., Dong, X., Zhang, P., He, C., Wang, J., Zhao, F., Lin, D.: Sharegpt4v: Improving large multi-modal models with better captions. In: European Conference on Computer Vision. pp. 370–387. Springer (2024)

[17] Chen, L., Li, J., Dong, X., Zhang, P., Zang, Y., Chen, Z., Duan, H., Wang, J., Qiao, Y., Lin, D., et al.: Are we on the right way for evaluating large vision-language models? Advances in Neural Information Processing Systems **37**, 27056–27087 (2024)

[18] Chen, Y., Wang, X., Peng, H., Ji, H.: Solo: A single transformer for scalable vision-language modeling. arXiv preprint arXiv:2407.06438 (2024)

[19] Chen, Z., Wang, W., Cao, Y., Liu, Y., Gao, Z., Cui, E., Zhu, J., Ye, S., Tian, H., Liu, Z., et al.: Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. arXiv preprint arXiv:2412.05271 (2024)

[20] Chen, Z., Wu, J., Wang, W., Su, W., Chen, G., Xing, S., Zhong, M., Zhang, Q., Zhu, X., Lu, L., et al.: Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 24185–24198 (2024)

[21] Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., et al.: Training verifiers to solve math word problems, 2021. URL https://arxiv. org/abs/2110.14168 **9** (2021)

[22] Cui, Y., Chen, H., Deng, H., Huang, X., Li, X., Liu, J., Liu, Y., Luo, Z., Wang, J., Wang, W., et al.: Emu3. 5: Native multimodal models are world learners. arXiv preprint arXiv:2510.26583 (2025)

[23] Dai, W., Lee, N., Wang, B., Yang, Z., Liu, Z., Barker, J., Rintamaki, T., Shoeybi, M., Catanzaro, B., Ping, W.: Nvlm: Open frontier-class multimodal llms. arXiv preprint arXiv:2409.11402 (2024)

[24] Das, A., Kottur, S., Gupta, K., Singh, A., Yadav, D., Moura, J.M., Parikh, D., Batra, D.: Visual dialog. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 326–335 (2017)

[25] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)

[26] Deniz, O.F., Mao, R., Li, R., Tian, Y., Khan, L.: Vision token reduction via attention-driven self-compression for efficient multimodal large language models. In: 2025 IEEE International Conference on Big Data (BigData). pp. 4544–4551. IEEE (2025)

[27] Diao, H., Cui, Y., Li, X., Wang, Y., Lu, H., Wang, X.: Unveiling encoder-free vision-language models. Advances in Neural Information Processing Systems **37**, 52545–52567 (2024)

[28] Diao, H., Li, M., Wu, S., Dai, L., Wang, X., Deng, H., Lu, L., Lin, D., Liu, Z.: From pixels to words–towards native vision-language primitives at scale. arXiv preprint arXiv:2510.14979 (2025)

[29] Dong, S., Hu, J., Zhang, M., Yin, M., Fu, Y., Qian, Q.: Mmtok: Multimodal coverage maximization for efficient inference of vlms. arXiv preprint arXiv:2508.18264 (2025)

[30] Dong, X., Zhang, P., Zang, Y., Cao, Y., Wang, B., Ouyang, L., Zhang, S., Duan, H., Zhang, W., Li, Y., et al.: Internlm-xcomposer2-4khd: A pioneering large vision-language model handling resolutions from 336 pixels to 4k hd. Advances in Neural Information Processing Systems **37**, 42566–42592 (2024)

[31] Duan, H., Yang, J., Qiao, Y., Fang, X., Chen, L., Liu, Y., Dong, X., Zang, Y., Zhang, P., Wang, J., et al.: Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In: Proceedings of the 32nd ACM international conference on multimedia. pp. 11198–11201 (2024)

[32] Fu, C., Chen, P., Shen, Y., Qin, Y., Zhang, M., Lin, X., Yang, J., Zheng, X., Li, K., Sun, X., et al.: Mme: A comprehensive evaluation benchmark for multimodal large language models. arXiv preprint arXiv:2306.13394 (2023)

[33] Gao, J., Pi, R., Zhang, J., Ye, J., Zhong, W., Wang, Y., Hong, L., Han, J., Xu, H., Li, Z., et al.: G-llava: Solving geometric problem with multi-modal large language model. arXiv preprint arXiv:2312.11370 (2023)

[34] Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., Parikh, D.: Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 6904–6913 (2017)

[35] Guan, T., Liu, F., Wu, X., Xian, R., Li, Z., Liu, X., Wang, X., Chen, L., Huang, F., Yacoob, Y., et al.: Hallusionbench: an advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 14375–14385 (2024)

[36] Gupta, A., Vedaldi, A., Zisserman, A.: Synthetic data for text localisation in natural images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2315–2324 (2016)

[37] Hudson, D.A., Manning, C.D.: Gqa: A new dataset for real-world visual reasoning and compositional question answering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6700–6709 (2019)

[38] Johnson, J., Hariharan, B., Van Der Maaten, L., Fei-Fei, L., Lawrence Zitnick, C., Girshick, R.: Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2901–2910 (2017)

[39] Kafle, K., Price, B., Cohen, S., Kanan, C.: Dvqa: Understanding data visualizations via question answering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5648–5656 (2018)

[40] Kahou, S.E., Michalski, V., Atkinson, A., Kádár, Á., Trischler, A., Bengio, Y.: Figureqa: An annotated figure dataset for visual reasoning. arXiv preprint arXiv:1710.07300 (2017)

[41] Kantharaj, S., Leong, R.T., Lin, X., Masry, A., Thakkar, M., Hoque, E., Joty, S.: Chart-to-text: A large-scale benchmark for chart summarization. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 4005–4023 (2022)

[42] Kembhavi, A., Salvato, M., Kolve, E., Seo, M., Hajishirzi, H., Farhadi, A.: A diagram is worth a dozen images. In: European conference on computer vision. pp. 235–251. Springer (2016)

[43] Kembhavi, A., Seo, M., Schwenk, D., Choi, J., Farhadi, A., Hajishirzi, H.: Are you smarter than a sixth grader? textbook question answering for multimodal machine comprehension. In: Proceedings of the IEEE Conference on Computer Vision and Pattern recognition. pp. 4999–5007 (2017)

[44] Kim, Y., Yim, M., Song, K.Y.: Tablevqa-bench: A visual question answering benchmark on multiple table domains. arXiv preprint arXiv:2404.19205 (2024)

[45] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4015–4026 (2023)

[46] Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.J., Shamma, D.A., et al.: Visual genome: Connecting language and vision using crowdsourced dense image annotations. International journal of computer vision **123**(1), 32–73 (2017)

[47] Laurençon, H., Saulnier, L., Tronchon, L., Bekman, S., Singh, A., Lozhkov, A., Wang, T., Karamcheti, S., Rush, A., Kiela, D., et al.: Obelics: An open web-scale filtered dataset of interleaved image-text documents. Advances in Neural Information Processing Systems **36**, 71683–71702 (2023)

[48] Lei, W., Wang, J., Wang, H., Li, X., Liew, J.H., Feng, J., Huang, Z.: The scalability of simplicity: Empirical analysis of vision-language learning with a single transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 20758–20769 (2025)

[49] Lerner, P., Ferret, O., Guinaudeau, C., Le Borgne, H., Besançon, R., Moreno, J.G., Lovón Melgarejo, J.: Viquae, a dataset for knowledge-based visual question answering about named entities. In: Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval. pp. 3108–3120 (2022)

[50] Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., et al.: Llava-onevision: Easy visual task transfer. arXiv preprint arXiv:2408.03326 (2024)

[51] Li, B., Wang, R., Wang, G., Ge, Y., Ge, Y., Shan, Y.: Seed-bench: Benchmarking multimodal llms with generative comprehension. arXiv preprint arXiv:2307.16125 (2023)

[52] Li, H., Peng, X., Wang, Y., Peng, Z., Chen, X., Weng, R., Wang, J., Cai, X., Dai, W., Xiong, H.: Onecat: Decoder-only auto-regressive model for unified understanding and generation. arXiv preprint arXiv:2509.03498 (2025)

[53] Li, W., Yuan, Y., Liu, J., Tang, D., Wang, S., Qin, J., Zhu, J., Zhang, L.: Tokenpacker: Efficient visual projector for multimodal llm. International Journal of Computer Vision **133**(10), 6794–6812 (2025)

[54] Li, X., Tu, H., Hui, M., Wang, Z., Zhao, B., Xiao, J., Ren, S., Mei, J., Liu, Q., Zheng, H., Zhou, Y., Xie, C.: What if we recaption billions of web images with llama-3? arXiv preprint arXiv:2406.08478 (2024)

[55] Li, Y., Du, Y., Zhou, K., Wang, J., Zhao, W.X., Wen, J.R.: Evaluating object hallucination in large vision-language models. In: Proceedings of the 2023 conference on empirical methods in natural language processing. pp. 292–305 (2023)

[56] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)

[57] Lindström, A.D., Abraham, S.S.: Clevr-math: A dataset for compositional language, visual and mathematical reasoning. arXiv preprint arXiv:2208.05358 (2022)

[58] Liu, H., Li, C., Li, Y., Lee, Y.J.: Improved baselines with visual instruction tuning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 26296–26306 (2024)

[59] Liu, H., Li, C., Li, Y., Li, B., Zhang, Y., Shen, S., Lee, Y.J.: Llavanext: Improved reasoning, ocr, and world knowledge (2024)

[60] Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. Advances in neural information processing systems **36**, 34892–34916 (2023)

[61] Liu, Y., Duan, H., Zhang, Y., Li, B., Zhang, S., Zhao, W., Yuan, Y., Wang, J., He, C., Liu, Z., et al.: Mmbench: Is your multi-modal model an all-around player? In: European conference on computer vision. pp. 216–233. Springer (2024)

[62] Lu, P., Bansal, H., Xia, T., Liu, J., Li, C., Hajishirzi, H., Cheng, H., Chang, K.W., Galley, M., Gao, J.: Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. arXiv preprint arXiv:2310.02255 (2023)

[63] Lu, P., Gong, R., Jiang, S., Qiu, L., Huang, S., Liang, X., Zhu, S.C.: Inter-gps: Interpretable geometry problem solving with formal language and symbolic reasoning. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 6774–6786 (2021)

[64] Lu, P., Mishra, S., Xia, T., Qiu, L., Chang, K.W., Zhu, S.C., Tafjord, O., Clark, P., Kalyan, A.: Learn to explain: Multimodal reasoning via thought chains for science question answering. Advances in neural information processing systems **35**, 2507–2521 (2022)

[65] Lu, P., Qiu, L., Chang, K.W., Wu, Y.N., Zhu, S.C., Rajpurohit, T., Clark, P., Kalyan, A.: Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. arXiv preprint arXiv:2209.14610 (2022)

[66] Lu, P., Qiu, L., Chen, J., Xia, T., Zhao, Y., Zhang, W., Yu, Z., Liang, X., Zhu, S.C.: Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning. arXiv preprint arXiv:2110.13214 (2021)

[67] Luo, W., Chen, P., Huang, X., Huang, L.: Vitcop: Accelerating large vision-language models via visual and textual semantic collaborative pruning. arXiv preprint arXiv:2601.17818 (2026)

[68] Ma, Y., Du, L., Shen, X., Chen, S., Li, P., Ren, Q., Ma, L., Dai, Y., Liu, P., Yan, J.: One rl to see them all: Visual triple unified reinforcement learning. arXiv preprint arXiv:2505.18129 (2025)

[69] Mao, W., Yang, Z., Shou, M.Z.: Unimod: Efficient unified multimodal transformers with mixture-of-depths. arXiv preprint arXiv:2502.06474 (2025)

[70] Marino, K., Rastegari, M., Farhadi, A., Mottaghi, R.: Ok-vqa: A visual question answering benchmark requiring external knowledge. In: Proceedings of the IEEE/cvf conference on computer vision and pattern recognition. pp. 3195–3204 (2019)

[71] Masry, A., Do, X.L., Tan, J.Q., Joty, S., Hoque, E.: Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In: Findings of the association for computational linguistics: ACL 2022. pp. 2263–2279 (2022)

[72] Mathew, M., Bagal, V., Tito, R., Karatzas, D., Valveny, E., Jawahar, C.: Infographicvqa. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1697–1706 (2022)

[73] Mathew, M., Karatzas, D., Jawahar, C.: Docvqa: A dataset for vqa on document images. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 2200–2209 (2021)

[74] Meng, C., Feng, P., Pang, J.J., Liao, C.T., Fu, R., Kang, Z., Gan, Z., Ouyang, C.: Adaptive hierarchical representation alliance for multimodal learning. arXiv preprint arXiv:2608.22863 (2026), `https://arxiv.org/abs/2608.22863`

[75] Methani, N., Ganguly, P., Khapra, M.M., Kumar, P.: Plotqa: Reasoning over scientific plots. In: Proceedings of the ieee/cvf winter conference on applications of computer vision. pp. 1527–1536 (2020)

[76] Mishra, A., Shekhar, S., Singh, A.K., Chakraborty, A.: Ocr-vqa: Visual question answering by reading text in images. In: 2019 international conference on document analysis and recognition (ICDAR). pp. 947–952. IEEE (2019)

[77] Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al.: LAION-5B: An open large-scale dataset for training next generation image-text models. In: Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track (2022)

[78] Schuhmann, C., Köpf, A., Vencu, R., Coombes, T., Beaumont, R.: LAION-COCO: 600M synthetic captions from LAION2B-EN. `https://laion.ai/blog/laion-coco/` (2022), lAION Blog. Re-released as ReLAION-COCO at `https://huggingface.co/datasets/laion/relaion-coco`

[79] Schwenk, D., Khandelwal, A., Clark, C., Marino, K., Mottaghi, R.: A-okvqa: A benchmark for visual question answering using world knowledge. In: European conference on computer vision. pp. 146–162. Springer (2022)

[80] Serra, A.P., Ortu, F., Panizon, E., Valeriani, L., Basile, L., Ansuini, A., Doimo, D., Cazzaniga, A.: The narrow gate: Localized image-text communication in native multimodal models. arXiv preprint arXiv:2412.06646 (2024)

[81] Shaharabany, T., Tewel, Y., Wolf, L.: What is where by looking: Weakly-supervised open-world phrase-grounding without text inputs. Advances in Neural Information Processing Systems **35**, 28222–28237 (2022)

[82] Shang, Y., Cai, M., Xu, B., Lee, Y.J., Yan, Y.: Llava-prumerge: Adaptive token reduction for efficient large multimodal models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 22857–22867 (2025)

[83] Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y., Wu, Y., et al.: Deepseekmath: Pushing the limits of mathematical reasoning in open language models. arXiv preprint arXiv:2402.03300 (2024)

[84] Sidorov, O., Hu, R., Rohrbach, M., Singh, A.: Textcaps: a dataset for image captioning with reading comprehension. In: European conference on computer vision. pp. 742–758. Springer (2020)

[85] Singh, A., Natarajan, V., Shah, M., Jiang, Y., Chen, X., Batra, D., Parikh, D., Rohrbach, M.: Towards vqa models that can read. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8317–8326 (2019)

[86] Song, J., Zhou, F., Qu, J., Li, W.J., Wu, T., Xue, G., Zhao, Z., Wei, D., Lu, Y., Na, B.: Evocomp: Learning visual token compression for multimodal large language models via semantic-guided evolutionary labeling. arXiv preprint arXiv:2604.17087 (2026)

[87] Sun, K., Bai, Y., Qi, J., Hou, L., Li, J.: Mm-math: Advancing multimodal math evaluation with process evaluation and fine-grained classification. In: Findings of the Association for Computational Linguistics: EMNLP 2024. pp. 1358–1375 (2024)

[88] Tan, Y., Sun, Y., Huang, S., Liu, H., Yu, G., Zhu, J., Deng, Y.: Idpruner: Harmonizing importance and diversity in visual token pruning for mllms. arXiv preprint arXiv:2602.13315 (2026)

[89] Tang, B., Boggust, A., Satyanarayan, A.: Vistext: A benchmark for semantically rich chart captioning. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 7268–7298 (2023)

[90] Team, C.: Chameleon: Mixed-modal early-fusion foundation models. arXiv preprint arXiv:2405.09818 (2024)

[91] Team, Q.: Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. arXiv preprint arXiv:2308.12966 (2023)

[92] Tong, E., Bai, Y., Zhu, Y., Jiang, J., Liu, X.: Focus-scan-refine: From human visual perception to efficient visual token pruning. arXiv preprint arXiv:2602.05809 (2026)

[93] Wang, J., Meng, L., Weng, Z., He, B., Wu, Z., Jiang, Y.G.: To see is to believe: Prompting gpt-4v for better visual instruction tuning. arXiv preprint arXiv:2311.07574 (2023)

[94] Wang, P., Bai, S., Tan, S., Wang, S., Fan, Z., Bai, J., Chen, K., Liu, X., Wang, J., Ge, W., et al.: Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. arXiv preprint arXiv:2409.12191 (2024)

[95] Wang, X., Zhang, X., Luo, Z., Sun, Q., Cui, Y., Wang, J., Zhang, F., Wang, Y., Li, Z., Yu, Q., et al.: Emu3: Next-token prediction is all you need. arXiv preprint arXiv:2409.18869 (2024)

[96] Wang, X., Liu, Y., Shen, C., Ng, C.C., Luo, C., Jin, L., Chan, C.S., Hengel, A.v.d., Wang, L.: On the general value of evidence, and bilingual scene-text visual question answering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10126–10135 (2020)

[97] Wang, Z., Sun, Y., Wang, H., Jing, B., Shen, X., Dong, X.L., Hao, Z., Xiong, H., Song, Y.: Reasoning-enhanced domain-adaptive pretraining of multimodal large language models for short video content governance. In: Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track. pp. 1104–1112 (2025)

[98] Wen, Z., Yang, B., Chen, S., Zhang, Y., Han, Y., Ke, J., Wang, C., Fu, Y., Zhao, J., Yao, J., et al.: Innovator-vl: A multimodal large language model for scientific discovery. arXiv preprint arXiv:2601.19325 (2026)

[99] Wu, H., Fan, Y., Dai, J., Tong, J., Ma, Y., Shen, X.: Hidrop: Hierarchical vision token reduction in mllms via late injection, concave pyramid pruning, and early exit. arXiv preprint arXiv:2602.23699 (2026)

[100] Wu, Y., Zhang, Z., Chen, J., Tang, H., Li, D., Fang, Y., Zhu, L., Xie, E., Yin, H., Yi, L., et al.: Vila-u: a unified foundation model integrating visual understanding and generation. arXiv preprint arXiv:2409.04429 (2024)

[101] Xia, R., Zhang, B., Ye, H., Yan, X., Liu, Q., Zhou, H., Chen, Z., Ye, P., Dou, M., Shi, B., et al.: Chartx & chartvlm: A versatile benchmark and foundation model for complicated chart reasoning, 2025. URL https://arxiv. org/abs/2402.12185 **2**(3)

[102] Xie, C., Cai, H., Song, J., Li, J., Kong, F., Wu, X., Morimitsu, H., Yao, L., Wang, D., Leng, D., Zhang, B., Ji, X., Deng, Y.: Zero and R2D2: A large-scale Chinese cross-modal benchmark and A vision-language framework. arXiv preprint arXiv:2205.03860 (2022)

[103] Xie, J., Mao, W., Bai, Z., Zhang, D.J., Wang, W., Lin, K.Q., Gu, Y., Chen, Z., Yang, Z., Shou, M.Z.: Show-o: One single transformer to unify multimodal understanding and generation. arXiv preprint arXiv:2408.12528 (2024)

[104] Xing, L., Huang, Q., Dong, X., Lu, J., Zhang, P., Zang, Y., Cao, Y., He, C., Wang, J., Wu, F., et al.: Pyramiddrop: Accelerating your large vision-language models via pyramid visual redundancy reduction. arXiv preprint arXiv:2410.17247 (2024)

[105] Xu, H., Xie, S., Tan, X.E., Huang, P.Y., Howes, R., Sharma, V., Li, S.W., Ghosh, G., Zettlemoyer, L., Feichtenhofer, C.: Demystifying clip data. arXiv preprint arXiv:2309.16671 (2023)

[106] Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al.: Qwen3 technical report. arXiv preprint arXiv:2505.09388 (2025)

[107] Yang, S., Chen, Y., Tian, Z., Wang, C., Li, J., Yu, B., Jia, J.: Visionzip: Longer is better but not necessary in vision language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19792–19802 (2025)

[108] Ye, W., Wu, Q., Lin, W., Zhou, Y.: Fit and prune: Fast and training-free visual token pruning for multi-modal large language models. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 22128–22136 (2025)

[109] Yu, H., Li, W., Qu, X., Wang, S., Chen, J., Zhu, J.: Visiontrim: Unified vision token compression for training-free mllm acceleration. arXiv preprint arXiv:2601.22674 (2026)

[110] Yu, L., Poirson, P., Yang, S., Berg, A.C., Berg, T.L.: Modeling context in referring expressions. In: European conference on computer vision. pp. 69–85. Springer (2016)

[111] Yuan, Y., Ji, Y., Li, Z., Li, R.: Sp-mind: An autonomous reasoning agent for spatial proteomics analysis. arXiv preprint arXiv:2606.24235 (2026)

[112] Yue, X., Ni, Y., Zhang, K., Zheng, T., Liu, R., Zhang, G., Stevens, S., Jiang, D., Ren, W., Sun, Y., et al.: Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9556–9567 (2024)

[113] Zhang, E., Yu, F., Wu, A., Wen, Z., Yan, K., Ding, S., Qi, B., Zhang, L.: D^2pruner: Debiased importance and structural diversity for mllm token pruning. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 40, pp. 12412–12420 (2026)

[114] Zhang, Q., Liu, M., Li, L., Lu, M., Zhang, Y., Pan, J., She, Q., Zhang, S.: Beyond attention or similarity: Maximizing conditional diversity for token pruning in mllms. arXiv preprint arXiv:2506.10967 (2025)

[115] Zhang, W., Zhu, Z., Li, N., Tao, S., Liu, K., Liu, Y.: Adaptinfer: Adaptive token pruning for vision-language model inference with dynamical text guidance. arXiv preprint arXiv:2508.06084 (2025)

[116] Zhang, Y.F., Lu, X., Yin, S., Fu, C., Chen, W., Hu, X., Wen, B., Jiang, K., Liu, C., Zhang, T., et al.: Thyme: Think beyond images. arXiv preprint arXiv:2508.11630 (2025)

[117] Zhang, Y., Fan, C.K., Ma, J., Zheng, W., Huang, T., Cheng, K., Gudovskiy, D., Okuno, T., Nakata, Y., Keutzer, K., et al.: Sparsevlm: Visual token sparsification for efficient vision-language model inference. arXiv preprint arXiv:2410.04417 (2024)

[118] Zhao, S., Zhang, X., Guo, J., Hu, J., Duan, L., Fu, M., Chng, Y.X., Wang, G.H., Chen, Q.G., Xu, Z., et al.: Unified multimodal understanding and generation models: Advances, challenges, and opportunities. arXiv preprint arXiv:2505.02567 (2025)

[119] Zhu, J., Wang, W., Chen, Z., Liu, Z., Ye, S., Gu, L., Tian, H., Duan, Y., Su, W., Shao, J., et al.: Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. arXiv preprint arXiv:2504.10479 (2025)

[120] Zhu, Y., Groth, O., Bernstein, M., Fei-Fei, L.: Visual7w: Grounded question answering in images. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4995–5004 (2016)

[121] Zou, X., Lu, D., Wang, Y., Yan, Y., Lyu, Y., Zheng, X., Zhang, L., Hu, X.: Don't just chase" highlighted tokens" in mllms: Revisiting visual holistic context retention. arXiv preprint arXiv:2510.02912 (2025)

A Supplementary Material

In this supplementary material, we provide: (A) related work (§A.1); (B) theoretical analysis and proofs for SDIE and PACS (§B); (C) per-stage training results comparing original and compressed models (§C); (D) detailed Pareto front construction and compression configurations (§D); (E) additional qualitative case studies comparing PACE with transferred baselines (§E); (F) extended experimental results and ablations (§F); (G) model architecture details, training data composition, and hyperparameter configurations (§A.4, §A.2).

A.1 Related Work

Multimodal Large Language Models. Multimodal learning spans diverse application domains and also addresses cross modal semantic depth mismatch through AHRA's hierarchical expert selection [74] and short video content governance through reasoning enhanced domain adaptive pretraining [97]. SP-Mind automates spatial proteomics analysis through an autonomous reasoning agent [111]. PDU-Net combines path aggregation and dual attention for lung nodule segmentation [11]. Modular MLLMs connect an independent ViT with a language model. The LLaVA series [60; 58; 59], InternVL [20; 19; 119], Qwen-VL [91; 94], and LLaVA-OneVision [50; 50] show strong capabilities across visual question answering, video understanding, and related tasks, establishing the "ViT + projection + LLM" pipeline as the dominant paradigm [30; 23]. This design, however, couples an externally pretrained encoder with the language model, incurring representational misalignment at the modality boundary and extra deployment complexity. Native MLLMs instead discard the ViT and jointly model visual and linguistic sequences within a unified autoregressive Transformer, offering a simpler and more scalable formulation. Pioneered by Fuyu [7] and EVE [27], this direction is advanced by Chameleon [90], Emu3 [95], Show-o [103], VILA-U [100], Emu3.5 [22], SAIL [48], and NEO [28], collectively validating the paradigm [118]. Among them, NEO introduces the Pre-Buffer and Post-LLM layered design [28], UniMoD [69] explores a mixed-depth perspective for unified understanding and generation, and The Narrow Gate [80] reveals the localized nature of cross-modal information flow. Yet since every pixel patch is tokenized directly without a ViT's spatial downsampling, native MLLMs accumulate far longer visual sequences, and a dedicated token compression framework for them remains absent.

Visual Token Compression. Visual token compression reduces redundancy in the visual sequences of MLLMs and has yielded a rich body of methods under modular architectures. FastV [15] first prunes low-contribution tokens via text-to-vision attention. Subsequent work extends this along several axes: PyramidDrop [104], HiDrop [99], and AdaptInfer [115] introduce progressive or dynamically scheduled multi-stage compression; SparseVLM [117] and FitPrune [108] enable training-free acceleration; LLaVA-PruMerge [82] jointly prunes and merges; DivPrune [2], CDPruner [114], HoloV [121], and FSR [92] select tokens from a semantic-coverage view; VisionZip [107] and MMTok [29] cast coverage maximization as an objective; VisionTrim [109] unifies encoding- and decoding-stage compression; and IDPruner [88], AgilePruner [5], D^2Pruner [113], and ViTCoP [67] jointly optimize importance and diversity. Orthogonally, ADSC [26] applies uniform layer-wise downsampling, while EvoComp [86] trains a learned compressor via offline evolutionary search. Across these methods, importance signals are positional-agnostic or single-axis, and scheduling relies on layer-level heuristics rather than an explicit multi-objective trade-off. Although transplantable to native backbones (our baselines in Table 2), they neither exploit the decoupled multi-axis positional structure of native MLLMs nor jointly optimize performance, efficiency, and coverage. The proposed PACE is the first to frame native visual token compression as Pareto multi-objective optimization, exploiting multi-axis structure via SDIE and searching schedules via PACS.

A.2 Training Data and Hyperparameter Details

Pre-training. The model learns basic visual concepts from large-scale image-text paired data. The training corpus comprises approximately 518M image-text pairs: BLIP3o [14] (29M), COYO [10] (314M, 10% sampled), Recap-DataComp [54] (610M, 50% sampled), and Relaion-COCO [78; 77] (333M, 50% sampled). During this stage, only the Patch Embedding layers, Pre-Buffer, and the newly added H/W-directional QK linear layers and normalization parameters in the Post-LLM are optimized; the Post-LLM backbone parameters remain frozen.

Mid-training. This stage strengthens vision-language alignment and enhances recognition of high-resolution images, complex scenes, and dense OCR content, bridging the gap between coarse pre-training concepts and downstream instruction data. The training data comprises approximately 89M samples drawn from 8 heterogeneous sources, employing a concept-balanced sampling strategy based on MetaCLIP [105] with re-captioning by large VLMs to improve caption fidelity and reduce concept imbalance. In this stage all model parameters, including the Post-LLM backbone, are unfrozen and optimized end-to-end.

Supervised fine-tuning. High-quality multi-source instruction data is used to enhance instruction following and multi-turn conversational abilities. The data spans a broad range of task categories, including Captioning, General QA, Science, Chart, Mathematics, Knowledge, OCR, Document, Grounding, and Conversation, so as to cover both perception-heavy and reasoning-heavy scenarios (full source and proportion details in Table 8).

Reinforcement learning. A total of 12,000 samples are used for reinforcement learning, covering Document/OCR (16%), Chart Understanding (10%), General VQA (10%), Conversation/Instruction (12%), Mathematics/Geometry (15%), Captioning (10%), Grounding (5%), Science (20%), and Visual Reasoning (14%). To prioritize informative samples, we apply pass@6 filtering: each candidate is sampled 6 times, and only those with 1–5 correct responses (out of 6) are retained, while samples with 6/6 correct (trivial) or 0/6 correct (overly difficult) are discarded to maximize the learning signal (details in Table 9).

Table 6: **Pre-training data summary.** Total: ∼518M image-text pairs.

Dataset	Language	Size	Sampling Strategy	Notes
BLIP3o [14]	EN	29.4M	full sampling	High-quality synthetic captions
COYO [10]	EN	314M	partly sampling	Web-crawled, noisy
Recap-DataComp [54]	EN	610M	partly sampling	High-quality recaptions
Relaion-COCO [78; 77]	EN	333M	partly sampling	High quality recaptions
ImageNet-21K [25]	EN	10.2M	partly sampling	Classification labels as short captions
Total		**∼518M**		

Table 7: **Mid-training data summary.** Total: ∼89M samples.

Dataset	Language	Size	Sampling Strategy	Notes
ImageNet-21K [25]	EN, CH	1.6M	partly sampling	Classification labels as short captions
laionCN [3]	EN, CH	6.4M	partly sampling	English & Chinese web-crawled image–text pairs
COYO [10]	EN, CH	15.1M	partly sampling	Web-crawled, noisy
Recap-DataComp [54]	EN, CH	6.2M	partly sampling	High-quality recaptions, rewritten for dense semantics
SAM-1B [45]	EN, ZH	1.1M	partly sampling	Segmentation-grounded dense captions
MINT [4]	EN, CH	4.1M	partly sampling	Multimodal interleaved documents
Zero250M [102]	EN, CH	9.6M	partly sampling	Large-scale image–text corpus
Obelics [47]	EN, CH	45.4M	partly sampling	Interleaved image–text web documents
Total			**∼89M**	

Table 8: **SFT data summary.** Multi-source instruction data spanning 8 task categories. Total: 12M samples.

Task Category	Datasets
Captioning	TextCaps (en) [84], ShareGPT4V (en&zh) [16], COCO [56]
General QA	VQAv2 [34], GQA [37], OKVQA [70], A-OKVQA [79], VSR [81], VisualDialog [24], ViQuAE [49]
Science	AI2D [42], ScienceQA [64], TQA [43]
Chart	ChartQA [71], DVQA [39], PlotQA [75], IconQA [66], VisText [89], Chart-to-Text [41], FigureQA [40]
Mathematics	Geo170K [33], GeoQA+ [13], TabMWP [65], CLEVR-Math [57], Geometry3K [63]
OCR	OCRVQA [76], TextVQA [85], InfoVQA [72], DocVQA [73], ST-VQA [8], SynthText [36]
Grounding	RefCOCO/+/g [110], Visual Genome [46], Visual7W [120]
Conversation	LLaVA-150K [60], LVIS-Instruct4V [93], ALLaVA [12]

Table 9: **RL data category proportions.** Total: ∼12K samples.

Category	Sampling%	Representative Datasets
Document/OCR	14%	DocVQA [73], TextVQA [85], ESTVQA [96]
Chart Understanding	9%	ChartQA [71], ChartX [101], TableVQA [44]
General VQA	9%	VQAv2 [34], AOKVQA [79], GQA [37], POPE [55]
Conversation/Instruction	11%	Innovator-VL [98], Thyme [116]
Mathematics/Geometry	13%	GeoQA [13], GSM8K [21], TabMWP [65], MMMath [87]
Captioning	9%	CLEVR-Counting [38], TallyQA [1]
Grounding	4%	Orsta [68]
Science	18%	ScienceQA [64], SciVQA [9]
Visual Reasoning	13%	MMMU [112], AI2D [42]

A.3 Training Hyperparameters

Table 10 provides the complete hyperparameter configurations across all four training stages. All stages share the AdamW optimizer with identical momentum and epsilon settings, a cosine learning rate schedule, weight decay of 0.01 and bfloat16 numerical precision. The maximum sequence length is 16,384, 32,768, 32,768 and 8,192 tokens in four stages, respectively. The key differences across stages lie in the peak learning rate, which decreases from 8×10^{-4} in pre-training to 1×10^{-5} in RL as the model converges, and the global batch size, gradually reduced from 2,560 to 128 to match decreasing data volume and enable finer gradient updates. Input resolution is 256^2–1024^2 during pre-training and extends to 256^2–2048^2 in subsequent stages. A text-only data ratio of 0.3 is applied during pre-training and mid-training to preserve the backbone's general language capability.

Table 10: **Hyperparameter configurations for each training stage.**

Configuration	Pre-training	Mid-training	SFT	RL
Resolution	112^2~1024^2	256^2~1024^2	256^2~2048^2	256^2~2048^2
Optimizer	AdamW ($\beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 10^{-8}$)			
Weight decay	0.01			
LR schedule	cosine	cosine	cosine	cosine
Peak LR	5×10^{-4}	5×10^{-5}	5×10^{-5}	1×10^{-5}
Training steps	250K	46K	19K	2K
Warmup steps	1K	200	200	100
Max sequence length	16,384	32,768	32,768	8,192
Global batch size	2,560	1,920	640	128
Numerical precision	bfloat16			

A.4 Model Architecture Details

Our native backbone, which we refer to as "Ours" throughout the paper, is a ViT-free multimodal large language model built on Qwen3-1.7B [106]. Unlike modular designs that route image features through a separate visual encoder before feeding them into the LLM, our model directly tokenizes raw image patches via a lightweight convolutional Patch Embedding layer and processes them jointly with text tokens in a single autoregressive Transformer, eliminating the modality boundary entirely.

Overall structure. The model comprises three functional blocks. The **Patch & Word Embedding** front-end converts each 32×32 image patch into a visual token through a convolutional Patch Embedding layer, which is then downsampled by a Dense Embedding module to match the LLM hidden dimension d=2048; text tokens are embedded via the standard word embedding table shared with the LLM. The **Pre-Buffer** (Initial Fusion Buffer, IFB) consists of the first L_{buf}=4 Transformer layers, where cross-modal semantic alignment and Native-RoPE position registration take place; within the IFB, visual token discriminability remains near zero (see probing analysis in §2.1), and no compression is applied. The **Post-LLM body** comprises the remaining L_{post}=36 layers, which inherit the pre-trained Qwen3-1.7B weights and are augmented with additional H/W-directional QK

projections for spatially decoupled attention; PACE compression stages are placed within this body. The total depth is $L_{\text{buf}} + L_{\text{post}} = 40$ layers, with $\sim$2.2B parameters.

Attention mechanism. We adopt a *mixed* masking strategy: visual tokens attend *bidirectionally* within the same image, enabling full spatial context exchange, while text tokens follow the standard *autoregressive* causal mask. Cross-modal attention is *causal*—text tokens can attend to preceding visual tokens, but not vice versa. This is implemented via FlexAttention with a custom block mask that combines document-level isolation, per-image bidirectional attention, and text-side causal masking.

Native-RoPE. Each token is assigned a three-component position ID (t, h, w) corresponding to its sequence position, spatial height, and spatial width, respectively, thereby explicitly preserving both sequential and two-dimensional spatial information. Distinct rotary base frequencies (β_T=10^6, β_H=β_W=10^4) are applied independently to the three components. The QK projection is further extended with an additional H/W-directional linear layer: the original QK head dimension is 64 for the T component, while an additional 64-dimensional head is introduced and evenly divided into 32 dimensions for H and 32 for W, with each component equipped with its own RMS normalization. This structural design explicitly separates temporal and spatial positional interactions, providing the dimensional decoupling that SDIE (§2.2) directly exploits for direction-aware importance estimation.

Architecture configuration. Table 11 summarizes the full configuration.

Table 11: **Architecture configuration of our trained model.**

Configuration	**Ours-2B**
Backbone LLM	Qwen3-1.7B
Pre-Buffer layers (L_{buf})	4
Post-LLM layers (L_{post})	36
Total layers	40
Hidden dim. d	2048
Attention heads	16 (QK) / 8 (V)
Original QK head dim. (T)	64
Added QK head dim. (H+W)	32+32
FFN intermediate dim.	6144
Patch size	16×16
β_P / β_H / β_W	10^6 / 10^4 / 10^4
Attention type	Mixed (visual bidirectional + text autoregressive)
Total parameters	$\sim$2.2B

A.5 Evaluation Details

All benchmark evaluations are conducted using VLMEvalKit [31] with default settings (prompts, answer extraction, and scoring). We employ the `gpt-4o-1120` API as LLM judge. The greedy decoding (temperature $= 0$, `topK=1`) is used to ensure reproducibility. For MME, we report the sum of perception and reasoning scores. All experiments are run on a single H20-96G GPU.

B Theoretical Analysis

B.1 Optimality of SDIE Additive-Multiplicative Fusion

We provide a formal justification for the spatial-additive, positional-multiplicative fusion strategy in Eq. 5.

Theorem B.1 (Optimality of Additive-Multiplicative Fusion)**.** *Let $s_i^H, s_i^W \geq 0$ be the height and width saliency scores for visual token i, and $s_i^T \geq 0$ be its temporal (positional) relevance. Define the token importance as $s_i = f(s_i^H, s_i^W, s_i^T)$. Restrict to the class of separable functions $f = g(s_i^H, s_i^W) \cdot h(s_i^T)$ where g is bilinear in its arguments and h is linear. Under the following desiderata: i)* ***Uni-directional preservation****: if $s_i^H \gg 0$ but $s_i^W \approx 0$ (or vice versa), the token should be retained;*

ii) ***Global irrelevance suppression****: if $s_i^T \approx 0$, the token should be discarded regardless of spatial saliency; iii)* ***Monotonicity****: f is non-decreasing in each argument; then, under the additional parsimony restriction that discards the redundant cross term ($\gamma = 0$) and the Native-RoPE spatial-symmetry prior ($\alpha = \beta$), $f(s_i^H, s_i^W, s_i^T) = (s_i^H + s_i^W) \times s_i^T$ is the unique (up to positive scaling) form satisfying all three conditions within this function class.*

Proof. We proceed by elimination of alternative functional forms within the separable bilinear class.

Step 1: Spatial components must combine additively. Consider the general class of separable functions $f = g(s_i^H, s_i^W) \cdot h(s_i^T)$, where g is bilinear, *i.e.*, $g(s_i^H, s_i^W) = \alpha s_i^H + \beta s_i^W + \gamma s_i^H s_i^W$ with $\alpha, \beta, \gamma \geq 0$. Any purely multiplicative form ($\alpha = \beta = 0, \gamma > 0$) yields $g(a, 0) = g(0, b) = 0$ for all $a, b > 0$, violating desideratum (i), so we must have $\alpha, \beta > 0$. The cross term $\gamma s_i^H s_i^W$ vanishes whenever either component is zero, so it contributes nothing to uni-directional preservation (i); it can only rescale tokens that are already salient on both axes, which is not required by any of (i)–(iii). We therefore invoke the parsimony restriction stated in the theorem and set $\gamma = 0$, keeping the minimal form that meets the desiderata. This gives $g(s_i^H, s_i^W) = \alpha s_i^H + \beta s_i^W$ with $\alpha, \beta > 0$. We emphasize that uniqueness is asserted only under this parsimony restriction together with the symmetry prior below; without them, any $\gamma \geq 0$ and any $\alpha, \beta > 0$ also satisfy (i)–(iii).

Step 2: Temporal component must act as a multiplicative gate. Condition (ii) requires $f = 0$ whenever $s_i^T = 0$, regardless of s_i^H, s_i^W. This demands $h(0) = 0$. Together with monotonicity (iii) and non-negativity, and restricting h to be linear, we have $h(s_i^T) = s_i^T$ (up to positive scaling).

Step 3: Verification of monotonicity. For the resulting $f = (\alpha s_i^H + \beta s_i^W) \cdot s_i^T$:

$$\frac{\partial f}{\partial s_i^H} = \alpha \cdot s_i^T \geq 0, \quad \frac{\partial f}{\partial s_i^W} = \beta \cdot s_i^T \geq 0, \quad \frac{\partial f}{\partial s_i^T} = \alpha s_i^H + \beta s_i^W \geq 0. \tag{9}$$

All partial derivatives are non-negative, confirming condition (iii). Finally, invoking the spatial symmetry prior of Native-RoPE, which encodes height and width with identical base frequencies $\beta_H = \beta_W$ and hence assigns no a priori preference to either axis, forces $\alpha = \beta$; fixing the residual positive scale to $\alpha = \beta = 1$ yields $f = (s_i^H + s_i^W) \times s_i^T$. This pins down the unique representative of the family under the symmetry prior. □

Takeaway (Takeaway). Within the separable bilinear function class, the additive-multiplicative fusion is the unique form that simultaneously prevents uni-directional information loss (via addition) and suppresses globally irrelevant tokens (via multiplicative gating). This is consistent with the cross-dimensional heterogeneous redundancy pattern observed in our probing analysis.

B.2 Compression Error Bound and Pareto Optimality

Theorem B.2 (Compression Error Bound)**.** *Let π_θ be a native MLLM with L layers and n_v visual tokens per input. Let $\mathcal{V}^{(l)}$ denote the retained visual tokens after compression at layer $l \in \mathcal{S}$, with retention ratio r_l. Define the* ***compression error*** *as:*

$$\mathcal{E}_{comp} = \left\| logit_{full} - logit_{comp} \right\|_2, \tag{10}$$

where $logit_{full}$ and $logit_{comp}$ are the output logits with and without compression. Assume (A1) the value vectors are uniformly bounded, i.e., $\|\mathbf{v}_i^{(l)}\| \leq B_v$ for all i, l; (A2) perturbations introduced at different compression stages propagate approximately independently. Then:

$$\mathcal{E}_{comp} \leq \sum_{l \in \mathcal{S}} C_l \cdot (1 - r_l) \cdot \left(1 - \frac{\sigma_l}{\sigma_{\max}}\right), \tag{11}$$

where C_l is a layer-wise constant incorporating the Lipschitz constant of the Transformer block at layer l and the value norm bound B_v, σ_l is the discriminability at layer l, and $\sigma_{\max} = \max_{l \geq L_{buf}} \sigma_l$.

Proof. **Step 1: Per-layer perturbation analysis.** At each compression stage l, token removal introduces a perturbation $\Delta^{(l)}$ to the hidden states. By the Lipschitz continuity of the multi-head

attention and FFN sub-layers (with constant bounded by the spectral norms of W_Q, W_K, W_V, W_O and the FFN weight matrices):

$$\|\Delta^{(l+1)}\| \leq C'_l \cdot \left\| \sum_{i \notin \mathcal{V}^{(l+1)}} \alpha_{j,i}^{(l)} \mathbf{v}_i^{(l)} \right\|, \tag{12}$$

where $\alpha_{j,i}^{(l)}$ is the attention weight from text token j to visual token i at layer l, $\mathbf{v}_i^{(l)}$ is the value vector, and C'_l is the Lipschitz constant of the post-attention layers.

Step 2: Bounding dropped attention mass via importance scores. By the triangle inequality and the value norm bound (A1):

$$\left\| \sum_{i \notin \mathcal{V}^{(l+1)}} \alpha_{j,i}^{(l)} \mathbf{v}_i^{(l)} \right\| \leq B_v \cdot \sum_{i \notin \mathcal{V}^{(l+1)}} \alpha_{j,i}^{(l)}. \tag{13}$$

Under the SDIE scoring, the retained tokens are the top-$K^{(l)}$ by importance. The total attention mass on dropped tokens satisfies:

$$\sum_{i \notin \mathcal{V}^{(l+1)}} \alpha_{j,i}^{(l)} = (n_l - K^{(l)}) \cdot \bar{\alpha}_{\text{drop}}^{(l)} = (1 - r_l) \cdot n_l \cdot \bar{\alpha}_{\text{drop}}^{(l)}, \tag{14}$$

where $n_l = |\mathcal{V}^{(l)}|$ and $\bar{\alpha}_{\text{drop}}^{(l)}$ is the average attention weight of dropped tokens. When discriminability σ_l is high, the SDIE importance scores effectively separate salient from non-salient tokens, ensuring $\bar{\alpha}_{\text{drop}}^{(l)} \ll \bar{\alpha}_{\text{retain}}^{(l)}$.

We note one subtlety omitted above: after tokens are dropped, the softmax over the retained set is re-normalized, so the retained weights are rescaled by $Z_l = 1/\sum_{i \in \mathcal{V}^{(l+1)}} \alpha_{j,i}^{(l)} = 1/(1 - \sum_{i \notin \mathcal{V}^{(l+1)}} \alpha_{j,i}^{(l)})$. Since the dropped mass is small under high-discriminability ranking, $Z_l = 1 + O(1 - r_l)$, and the induced perturbation is dominated (to first order) by the dropped-mass term already bounded above; the re-normalization contributes only a higher-order $O((1 - r_l)^2)$ correction, which we absorb into C_l. The bound below is therefore stated to first order in the dropped mass.

Step 3: Discriminability correction factor. The quality of importance ranking depends on discriminability σ_l. When $\sigma_l \approx 0$ (within IFB), all tokens have near-identical importance scores and any ranking is essentially random; when σ_l is high, SDIE reliably assigns low scores to non-salient tokens. We model this relationship as:

$$\bar{\alpha}_{\text{drop}}^{(l)} \leq \bar{\alpha}_{\text{all}}^{(l)} \cdot \left(1 - \frac{\sigma_l}{\sigma_{\max}}\right), \tag{15}$$

where $\bar{\alpha}_{\text{all}}^{(l)} = 1/n_l$. This is an idealized monotone model: the correction factor interpolates between worst-case random dropping ($\sigma_l = 0$, factor $= 1$) and the ideal limit of perfect ranking ($\sigma_l = \sigma_{\max}$, factor $\to 0$). In practice, the bound remains informative because the factor only approaches zero at the deepest layers where SDIE ranking quality is empirically strongest. Absorbing B_v, C'_l, and $n_l \cdot \bar{\alpha}_{\text{all}}^{(l)}$ into a single constant $C_l = C'_l \cdot B_v$:

$$\|\Delta^{(l+1)}\| \leq C_l \cdot (1 - r_l) \cdot \left(1 - \frac{\sigma_l}{\sigma_{\max}}\right). \tag{16}$$

Step 4: Aggregation across stages. Under assumption (A2) that perturbations from different compression stages propagate approximately independently, summing the per-stage contributions via the triangle inequality yields:

$$\mathcal{E}_{\text{comp}} \leq \sum_{l \in \mathcal{S}} C_l \cdot (1 - r_l) \cdot \left(1 - \frac{\sigma_l}{\sigma_{\max}}\right). \tag{17}$$

□

Corollary B.3 (Pareto Schedule Minimizes the Bound)**.** *Consider a fixed total retention budget $R = \prod_{l \in \mathcal{S}} r_l$, equivalently the additive log-budget $\sum_{l \in \mathcal{S}} \log r_l = \log R$. Minimizing the upper bound in Eq. 11 subject to this constraint assigns higher retention to low-discriminability layers and lower retention to high-discriminability layers. Concretely, in the small-compression regime where*

$1 - r_l$ *is small we have* $\log r_l \approx -(1 - r_l)$*, so the constraint becomes* $\sum_l (1 - r_l) \approx -\log R$ *and the objective* $\sum_l C_l(1 - r_l)(1 - \sigma_l/\sigma_{\max})$ *is linear in the drop fractions* $\{1 - r_l\}$*. Minimizing a linear objective over the simplex* $\sum_l (1 - r_l) = -\log R$ *concentrates the available drop budget on the stage with the smallest coefficient* $C_l(1 - \sigma_l/\sigma_{\max})$*, i.e., the deepest, highest-discriminability layer. When per-stage caps* $1 - r_l \leq 1 - r_{\min}$ *are imposed to keep each stage's perturbation within tolerance, the minimizer fills the drop budget in increasing order of* $C_l(1 - \sigma_l/\sigma_{\max})$*, which orders retention as*

$$r^*_{l_1} \geq r^*_{l_2} \geq \cdots \geq r^*_{l_L} \quad \textit{whenever} \quad \sigma_{l_1} \leq \sigma_{l_2} \leq \cdots \leq \sigma_{l_L}, \tag{18}$$

i.e., stages with higher discriminability (larger σ_l*) tolerate more aggressive compression (lower* r_l*).*

Takeaway (Takeaway). The bound formally justifies two empirical observations: (1) compression should be initiated only after the IFB where $\sigma_l \approx 0$, and (2) the Pareto-adaptive schedule ($r_{l_1} > r_{l_2} > r_{l_3}$ aligned with increasing discriminability) systematically outperforms uniform compression at the same total retention.

B.3 Information-Theoretic Justification of Spatial Decoupling

Proposition B.4 (Native-RoPE Component Independence)**.** *Let* s_i^H, s_i^W *be the height and width saliency scores defined in Eq. 4. Under the Native-RoPE encoding with independent base frequencies* $\beta_T, \beta_H, \beta_W$ *and non-overlapping head-dimension partitions, the height and width components are computed from disjoint subspaces of the QK head dimension* d_h*. This structural independence implies that, in the absence of learned inter-subspace correlations,* s_i^H *and* s_i^W *are expected to exhibit low mutual correlation.*

Justification. Recall from Eq. 2 that the Query and Key vectors are partitioned as $\boldsymbol{q}_i = [\boldsymbol{q}_{i,t};\ \boldsymbol{q}_{i,h};\ \boldsymbol{q}_{i,w}]$ and $\boldsymbol{k}_j = [\boldsymbol{k}_{j,t};\ \boldsymbol{k}_{j,h};\ \boldsymbol{k}_{j,w}]$, where the h and w components occupy disjoint dimension ranges equipped with independent rotary frequencies Θ_H and Θ_W. Since $s_i^H = \frac{1}{|\mathcal{T}|} \sum_{j \in \mathcal{T}} \boldsymbol{q}_{j,h} \cdot \boldsymbol{k}_{i,h}^\top$ depends only on the h-subspace and s_i^W depends only on the w-subspace, the two scores share no parameters or input dimensions at initialization. While training may introduce indirect correlations through the shared loss, the disjoint subspace structure provides a strong inductive bias toward independence. □

Empirical verification. On our calibration set of 1,000 images, the average absolute Pearson correlation between (s_i^H, s_i^W) pairs is $\bar{r} = 0.073 \pm 0.021$ across all post-IFB layers. This near-zero correlation confirms that the height and width saliency scores capture independent redundancy modes in practice, validating the additive fusion in SDIE.

B.4 Finite-Grid Approximation Quality for Pareto Front

Proposition B.5 (Grid Approximation Bound)**.** *Let* $\mathcal{P}^*$ *be the true continuous Pareto-optimal set and* $\hat{\mathcal{P}}^*_\delta$ *be the Pareto set obtained from grid enumeration with step size* δ*. Then for any* $\boldsymbol{r}^* \in \mathcal{P}^*$*, there exists* $\hat{\boldsymbol{r}} \in \hat{\mathcal{P}}^*_\delta$ *such that:*

$$\|\boldsymbol{r}^* - \hat{\boldsymbol{r}}\|_\infty \leq \frac{\delta}{2}, \quad \textit{and} \quad \forall\, k:\ |\mathcal{O}_k(\boldsymbol{r}^*) - \mathcal{O}_k(\hat{\boldsymbol{r}})| \leq L_k \cdot \frac{\delta\sqrt{|\mathcal{S}|}}{2}, \tag{19}$$

where L_k *is the Lipschitz constant of objective* $\mathcal{O}_k$*. For our setting (*$\delta = 0.05$*,* $|\mathcal{S}| = 3$*), the maximum per-objective approximation error is bounded by* $0.043 \cdot L_k$*.*

This bound confirms that our discrete grid search with 4,913 candidates provides a close approximation to the continuous Pareto front. One caveat is that the nearest grid point $\hat{\boldsymbol{r}}$ to a continuous optimum $\boldsymbol{r}^*$ need not itself survive Pareto filtering: it may be dominated by another grid point that is better on all objectives. In that case the dominating point is at least as good on every $\mathcal{O}_k$, so replacing $\hat{\boldsymbol{r}}$ by the retained dominator only tightens the per-objective gap $|\mathcal{O}_k(\boldsymbol{r}^*) - \mathcal{O}_k(\cdot)|$ in the direction of improvement; the stated bound therefore continues to hold for the point actually returned by the search (the retained dominator), and the guarantee is on objective values rather than on the retention of any specific grid coordinate. In practice, the Lipschitz constants L_k are moderate: the performance objective varies smoothly with retention ratio (empirically $L_{\text{perf}} \approx 15$ benchmark points per unit retention change on our calibration set), yielding a maximum approximation error of $\sim$0.65 points, which is negligible relative to inter-benchmark performance variance.

C Per-Stage Training Results

C.1 Stage-wise Performance: Original vs. Compressed

Table 12 reports the performance of the model after each training stage under three conditions: no compression (full tokens), 50% token retention, and 25% token retention. Since the model lacks instruction-following ability after pre-training, benchmark evaluation is not meaningful at that stage; we therefore report results starting from mid-training. This reveals how the model's tolerance to compression evolves throughout training.

Table 12: **Per-stage benchmark performance: original vs. compressed.** We report results after each training stage (starting from mid-training) on eight benchmarks. Pre-training results are omitted because the model has not yet acquired instruction-following ability at that stage. **Green**: improvement over the uncompressed baseline of that stage; red: degradation.

Stage	Retention	MMB	AI2D	MME	HalBench	RWQA	MMStar	SEED	MathVista
Stage 2: Mid-training (∼89M samples)									
Mid-train	100%	58.4	69.3	1782	35.6	52.1	43.8	62.4	42.1
	50%	-2.1% (57.2)	**+0.4%** (69.6)	-1.3% (1759)	-2.8% (34.6)	-1.5% (51.3)	-3.4% (42.3)	-1.0% (61.8)	-4.5% (40.2)
	25%	-5.8% (55.0)	-2.7% (67.4)	-4.2% (1707)	-7.3% (33.0)	-4.6% (49.7)	-8.2% (40.2)	-4.8% (59.4)	-11.6% (37.2)
Stage 3: Supervised Fine-Tuning (∼12.8M samples)									
SFT	100%	67.8	79.2	2041	41.3	60.2	51.6	70.4	53.8
	50%	**+1.0%** (68.5)	**+2.8%** (81.4)	**+0.3%** (2047)	-0.7% (41.0)	**+0.7%** (60.6)	-2.3% (50.4)	**+0.4%** (70.7)	-3.2% (52.1)
	25%	-0.6% (67.4)	**+1.1%** (80.1)	-1.2% (2017)	-3.9% (39.7)	-1.8% (59.1)	-5.8% (48.6)	-1.6% (69.3)	-10.8% (48.0)
Stage 4: Reinforcement Learning (∼12K samples)									
RL (Final)	100%	69.7	80.8	2084	42.7	61.8	52.9	71.8	55.7
	50%	**+2.2%** (71.2)	**+4.0%** (84.0)	**+0.6%** (2098)	**+0.5%** (42.9)	**+1.3%** (62.6)	-1.7% (52.0)	**+1.0%** (72.5)	-2.5% (54.3)
	25%	**+1.4%** (70.7)	**+2.8%** (83.1)	**+0.4%** (2092)	-2.5% (41.6)	-1.0% (61.2)	-4.7% (50.4)	-0.6% (71.4)	-9.7% (50.3)

At the mid-training stage, the model has acquired basic vision-language alignment, and compression tolerance already shows encouraging signs across multiple benchmarks: at 50% retention, most benchmarks degrade by less than 3%, confirming that cross-modal alignment enables meaningful token discriminability. Nonetheless, at 25% retention degradation remains noticeable, reaching up to -11.6% on MathVista, indicating that mid-training alone does not yet produce sufficiently sharp importance differentiation to support more aggressive token pruning.

After SFT, the model begins to *benefit* from compression on several benchmarks (MMB $+1.0\%$, AI2D $+2.8\%$ at 50% retention), suggesting that instruction tuning reinforces the discriminability structure by sharpening task-relevant attention patterns. At the final RL stage, compression yields consistent gains on six of eight benchmarks at 50% retention and near-lossless performance at 25%, validating that RL further sharpens token importance differentiation. This progressive improvement in compression tolerance across training stages provides empirical support for our theoretical compression bound (Theorem B.2): as discriminability σ_l increases with training, the correction factor $(1 - \sigma_l/\sigma_{\max})$ decreases and the bound tightens.

C.2 Statistical Significance Analysis

Protocol. We select three benchmarks that jointly cover the core capability axes probed in the main experiments. **MMB** for general multimodal perception and compositional understanding, **AI2D** for structured reasoning that is sensitive to fine-grained spatial layout, and **HallBench** for hallucination robustness. This selection is determined *a priori* from the capability taxonomy used throughout the paper, not post-hoc from the measured variances. We draw 3 independently sampled calibration sets (different random seeds, each containing 1,000 images from the same distribution described in §3.1). For each calibration set, we run PACS end-to-end and obtain one searched schedule. Each schedule is then evaluated on **three representative benchmarks covering distinct capability domains**, i.e., MMB, AI2D and HallBench, at all three retention levels (25%, 50%, 75%), obtaining 27 independent measurements. We report mean $\pm$ standard deviation in Table 13.

Results. These results support the statistical significance of our main claims. Across all 27 measurements, the largest observed standard deviation is ***0.15***, indicating that PACS produces stable schedules across calibration resamples. On every (benchmark, retention) cell, the PACE-vs-baseline gap exceeds the observed std by a factor of at least ***14.7***$\times$.

Table 13: Statistical Significance Analysis on **3 independently sampled calibration sets** on MMB, AI2D and HallBench.

Configuration	Ret.	MMB	AI2D	HallBench
Full tokens	100%	69.7	80.8	42.7
PACE	25%	+1.4% (70.7 ± **0.06**)	+2.9% (83.1 ± **0.06**)	-2.5% (41.6 ± **0.1**)
	50%	+2.2% (71.2 ± **0.06**))	+4.0% (84.0 ± **0.1**))	+0.5% (42.9 ± **0.1**))
	75%	+3.2% (71.9 ± **0.15**))	+5.1% (84.9 ± **0.06**))	+3.9% (44.4 ± **0.06**))

D Pareto Front Construction and Compression Configurations

D.1 Pareto Front Construction Details

The Pareto front construction proceeds as follows. First, a single forward pass on 1,000 unlabeled calibration images extracts per-layer SDIE scores (s_i^T, s_i^H, s_i^W) for every layer $l \geq L_{\text{buf}}$. Second, per-layer discriminability σ_l is computed to guide compression-layer search. Third, the search jointly optimizes layer positions and per-stage retention ratios. Three ordered stages $\mathcal{S} = l_1, l_2, l_3$ ($l_1 < l_2 < l_3$) are selected from the 36 post-IFB layers, giving $\binom{36}{3} = 7{,}140$ configurations. Each retention ratio r_l is drawn from a shared 25-value set $\mathcal{G} = 0.10, 0.15, \ldots, 0.90 \cup k/9k = 1^8$, yielding $25^3 = 15{,}625$ ratio combinations and $\sim 1.12 \times 10^8$ total candidates. Fourth, for each candidate $(\mathcal{S}, \boldsymbol{r})$, the three objectives in Eq. 6 are evaluated on the calibration set: $\mathcal{O}$perf, the average benchmark-score degradation $\frac{1}{|\mathcal{B}|}\sum_b (S_b^{\text{full}} - S_b^{\text{comp}})$, estimated on the labeled subset with per-token output-logit KL divergence as an inexpensive unlabeled proxy during coarse traversal; $\mathcal{O}$eff $= 1 - \prod l \in \mathcal{S} r_l$; and $\mathcal{O}$cov from Eq. 8. Fifth, the non-dominated set $\mathcal{P}^*$ is extracted by standard Pareto dominance after discarding candidates with $\mathcal{O}\text{cov} > \tau_{\text{cov}}$. Guided by **Finding 1** (depth-progressive redundancy accumulation), we enforce $r_{l_1} > r_{l_2} > r_{l_3}$, substantially reducing the effective search space. Finally, for target retention r_{target}, the operating point is selected from $\mathcal{P}^*$ subject to $\prod_l r_l \leq r_{\text{target}}$ while minimizing $\mathcal{O}_{\text{perf}}$. The procedure completes in approximately 10 GPU-minutes on a single H20-96G.

D.2 Effect of Spatial Coverage Constraint

The spatial coverage constraint $\mathcal{O}_{\text{cov}}$ is an intrinsic component of the Pareto optimization formulation. Without this constraint, aggressive token dropping can concentrate retained tokens in salient regions while leaving large spatial gaps, which degrades performance on tasks requiring global spatial understanding such as scene description and spatial grounding. Table 14 quantifies the impact of this constraint.

Table 14: **Effect of spatial coverage constraint.** "w/ Cov." includes $\mathcal{O}_{\text{cov}}$ in Pareto optimization; "w/o Cov." removes it.

Config.	Ret.	RWQA	HalBench	SEED	MMB
w/ Cov. (Ours)	25%	**61.2**	**41.6**	**71.4**	**70.7**
w/o Cov.	25%	58.7	39.2	69.8	69.1
w/ Cov. (Ours)	50%	**62.6**	**42.9**	**72.5**	**71.2**
w/o Cov.	50%	61.4	42.1	72.0	70.8

The coverage constraint yields a 1.6 to 2.5 percentage point improvement at 25% retention on spatially sensitive benchmarks (RWQA, HalBench), validating its role as an intrinsic Pareto objective rather than a post-hoc correction. It maintains balanced spatial distribution among retained tokens, preventing loss of peripheral visual information critical for holistic scene understanding.

D.3 Compression Configuration Details

Table 15 summarizes the detailed compression configurations for the three retention targets. All configurations employ three sequential compression stages at layers L_{buf}, L_{buf}+9, and L_{buf}+16 (corresponding to layers 4, 13, and 20 in our 40-layer architecture), whose positions are determined by the structural discriminability curve: the first stage is placed at the inflection point

Table 15: **Compression configurations for three retention targets.** Per-stage ratios are from Pareto optimization over performance, efficiency, and spatial coverage. Stages are at layers 4, 13, 20.

Overall Ret.	r_{l_1} (L4)	r_{l_2} (L13)	r_{l_3} (L20)	Final Ret.%
75%	95%	90%	87%	74.36%
50%	85%	78%	75%	49.73%
25%	75%	66.7%	50%	25.01%

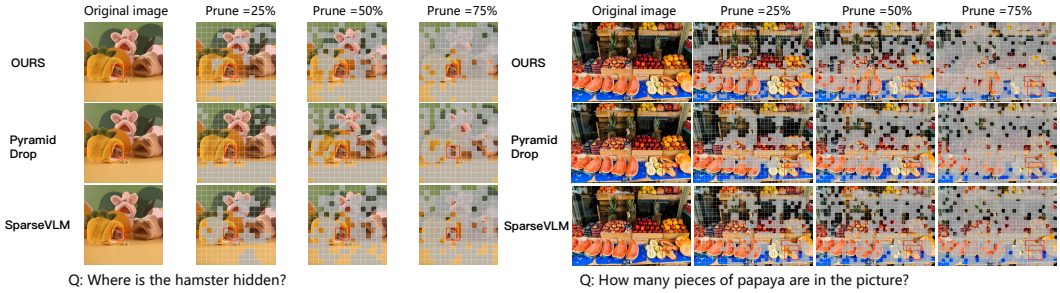


Figure 7: **Qualitative comparison of token retention (Part 1).** *Left* (object localization): our method retains tokens covering the camouflaged target and its spatial context, while PyramidDrop scatters tokens uniformly and SparseVLM over concentrates on a single cluster. *Right* (counting): our method distributes tokens across all fruit instances, whereas baselines exhibit concentration bias that drops peripheral ones.

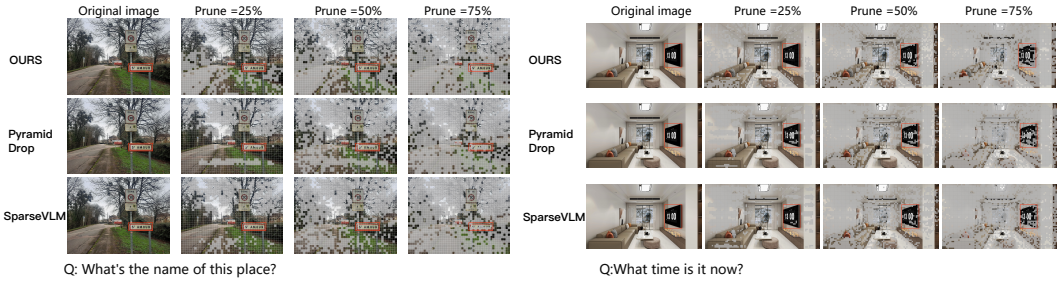


Figure 8: **Qualitative comparison of token retention (Part 2).** *Left* (text recognition): our method preserves sign text via s^H, while baselines introduce spatial gaps in lettering at high pruning. *Right* (fine grained detail): our method retains clock display tokens through s^P, whereas baselines progressively lose critical pixels.

where discriminability begins to rise, the second at the midpoint of the ascending segment, and the third on the high-discriminability plateau where tokens exhibit sufficient structural redundancy and aggressive compression can be safely applied without degrading performance.

The per-stage retention ratios follow a consistent pattern: $r_{l_1} > r_{l_2} > r_{l_3}$, meaning that earlier stages retain more tokens while later stages apply more aggressive compression. This pattern is directly predicted by Corollary B.3: since discriminability increases monotonically with depth, deeper layers with higher σ_l can tolerate lower retention ratios without exceeding the compression error bound. At 75% overall retention, the compression is mild across all stages (95%, 90%, 87%, product $\approx$74.4%), reflecting the limited redundancy that needs to be removed. At 50% overall retention, the schedule transitions to moderately aggressive compression (85%, 78%, 75%) while preserving near-lossless performance on most benchmarks. At 25% overall retention, the schedule becomes substantially more aggressive at the final stage (75%, 66.7%, 50%), exploiting the high discriminability at deep layers to safely discard half of the remaining tokens. In all cases the product of per-stage ratios satisfies the overall retention target $\prod_l r_l \leq r_{\text{target}}$.

E Additional Qualitative Case Studies

Figures 7 and 8 visualize the retained visual tokens of PACE alongside two representative transferred baselines, PyramidDrop† and SparseVLM†, at three pruning ratios (25%, 50%, 75%) on four diverse scenes. Colored patches indicate retained tokens, while grayed-out regions denote pruned tokens; red bounding boxes highlight task-critical regions.

Object localization and counting (Figure 7). In the "hamster" scene (left), PACE consistently retains tokens covering both the camouflaged hamster body and its surrounding plush context at all pruning levels, enabling correct localization even when 75% of visual tokens are removed.

Table 16: **Effect of number of compression stages.** Average over MMB, AI2D, HalBench, RWQA, and SEED at various retention levels.

Retention	$N = 1$	$N = 2$	$N = 3$ **(Ours)**	$N = 4$
75%	65.8	66.4	**67.3**	67.2
50%	63.2	65.1	**66.6**	66.5
25%	55.7	62.8	**65.6**	65.4

PyramidDrop scatters retained tokens relatively uniformly across the image without concentrating on the target region, while SparseVLM over focuses on a single salient cluster around the hamster head, losing the broader contextual cues that are necessary under aggressive pruning. In the "papaya" scene (right), accurate counting requires preserving all individual fruit instances that are distributed across the market stall. PACE distributes retained tokens broadly, covering every papaya group owing to the spatial coverage constraint in PACS, whereas both baselines exhibit spatial concentration bias: PyramidDrop clusters tokens in the upper center region, and SparseVLM over weights the foreground, causing peripheral papaya instances to be dropped once pruning reaches 50% and beyond.

Text recognition and fine-grained detail (Figure 8). For the road-sign scene (left, "What's the name of this place?"), PACE preserves tokens on the sign text "St Amour" across all pruning levels, benefiting from the height-directional saliency component s^H that naturally emphasizes horizontally aligned text regions. PyramidDrop retains scattered patches that partially overlap with the sign but introduces spatial gaps in the lettering area at 75% pruning, degrading readability. SparseVLM maintains reasonable sign coverage at low pruning but progressively loses critical letter regions at higher pruning ratios, as its single-score importance ranking cannot distinguish directional text saliency. In the indoor clock scene (right, "What time is it now?"), the task-critical region is the small digital display on the TV screen. PACE reliably retains tokens on the clock face across all pruning levels, benefiting from the positional multiplicative gate s^P that up-weights tokens receiving strong text-query attention. Both baselines retain the clock area at low pruning but progressively lose display pixels at 50% and 75%, leading to incorrect time readings.

These qualitative results corroborate the quantitative findings in Table 2: PACE's spatially decoupled importance estimation preserves task-relevant tokens more effectively than transferred modular baselines, particularly under aggressive compression.

F Extended Experimental Results

F.1 Extended Compression Stage Results

Table 16 provides the full numerical results underlying Figure 6, systematically comparing compression configurations with $N \in 1, 2, 3, 4$ stages across the evaluated benchmarks. Single-stage compression ($N = 1$) introduces overly abrupt information removal and consequently causes noticeable performance degradation, whereas progressively increasing the number of stages enables smoother token reduction. However, the additional benefit of a fourth stage ($N = 4$) over $N = 3$ remains marginal, confirming that three-stage compression provides the most effective balance between compression granularity, performance preservation, and structural complexity.

F.2 Robustness to Input Resolution

Table 17 evaluates PACE at resolutions from 256^2 to 2048^2, assessing how compression benefits scale with the number of visual tokens. Compression benefits increase with resolution. At 256^2 (64 tokens), compression causes mild degradation since there is little redundancy to exploit. At 2048^2 (4096 tokens), the gains are most pronounced (+1.7% MMB,

Table 17: **Resolution sensitivity under 25% token retention.** Performance on MMB and AI2D at different input resolutions.

Resolution	**MMB**		**AI2D**	
	Full	25% Ret.	Full	25% Ret.
256^2 (∼64 tokens)	61.2	59.8 (−2.3%)	70.1	69.5 (−0.9%)
512^2 (∼256 tokens)	66.4	67.1 (+1.1%)	76.5	78.2 (+2.2%)
1024^2 (∼1024 tokens)	69.7	70.7 (+1.4%)	80.8	83.1 (+2.8%)
2048^2 (∼4096 tokens)	70.3	71.5 (+1.7%)	81.2	84.0 (+3.4%)

Table 19: **Fine-grained retention sweep on PACE-2B.** Performance across eight benchmarks at retention levels from 10% to 90%.

Retention	MMB	AI2D	MME	HalBench	RWQA	MMStar	SEED	MathVista
100% (Full)	69.7	80.8	2084	42.7	61.8	**52.9**	71.8	55.7
90%	70.5	81.3	2118	42.8	61.7	**52.9**	72.1	55.8
80%	**72.0**	83.1	**2124**	43.5	62.0	52.8	72.7	56.0
75%	71.9	**84.9**	2120	**44.4**	62.4	52.6	**73.1**	56.4
70%	71.6	84.5	2112	43.8	62.5	52.3	72.9	**56.6**
60%	71.4	84.2	2103	43.1	**62.6**	52.1	72.7	55.0
50%	71.2	84.0	2098	42.9	**62.6**	52.0	72.5	54.3
40%	71.0	83.5	2095	42.1	61.8	51.2	72.0	52.1
30%	70.8	83.3	2093	41.8	61.4	50.8	71.6	50.9
25%	70.7	83.1	2092	41.6	61.2	50.4	71.4	50.3
20%	69.8	81.7	2052	40.2	60.1	48.9	70.2	47.1
10%	68.6	77.8	1872	35.4	55.2	42.7	65.1	38.6

$+3.4\%$ AI2D), confirming that native MLLMs exhibit increasing visual token redundancy at higher resolutions, precisely where PACE provides the greatest value.

F.3 Layer Position Selection

Table 18 evaluates different compression layer positions for visual tokens under the three-stage configuration. The optimal positions, layers 4, 13, 20 (i.e., L_{buf}, L_{buf}+9, L_{buf}+16 in the 40-layer architecture), align with the discriminability curve: the first stage at the inflection point, the second at the ascending segment, and the third on the high plateau.

Table 18: **Effect of compression layer positions on Ours-2B (MMB).** Layer indices are absolute positions in the 40-layer architecture. Three retention rate allocations are compared.

Layer Positions	90-56-50%	75-67-50%	60-60-70%
No compression		69.7	
L4-L13-L20	-3.7% (67.1)	+1.4% (70.7)	-1.2% (68.9)
L6-L14-L22	-2.3% (68.1)	-4.6% (66.5)	-4.2% (66.8)
L4-L16-L24	-0.7% (69.2)	-2.0% (68.3)	-4.2% (66.8)

F.4 Fine-Grained Retention Sweep

Table 19 provides a fine-grained retention sweep from 10% to 90%, revealing the performance-efficiency Pareto frontier at higher resolution. The model achieves a sweet spot at 50% to 75% retention where performance exceeds the uncompressed baseline on most benchmarks. Below 25%, performance remains near-lossless for most tasks except MathVista and MMStar. Below 20%, degradation accelerates, consistent with the compression bound in Theorem B.2 where the dropped attention mass begins to include semantically critical tokens.

F.5 Matched-Retention Batched-Decode Efficiency

Single-request (batch 1) decoding is dominated by weight loading rather than KV-cache reads, so at this operating point all methods share essentially the same decode latency and the runtime cannot separate compression schemes. To expose the regime where visual-token reduction actually matters, we report a *matched-retention* batched comparison against representative training-free baselines (SparseVLM, PyramidDrop) that, like PACE, keep only 256 visual tokens (25% retention). All numbers are measured on the same Ours-2B backbone and hardware. The five-benchmark average scores (Table 1) are 65.4 (Full), 64.6 (SparseVLM), 62.7 (PyramidDrop) and 65.6 (PACE).

We draw two observations. First, the batched-serving benefit is *generic to any* 25%*-retention method* rather than PACE-specific: at identical retention SparseVLM, PyramidDrop and PACE all reach $\approx$ 4.5 GB KV cache and $\approx$ 56–59 GB memory at batch 64 and continue to serve where the full model OOMs, so we do not claim the OOM-avoidance or batch-scaling envelope as a PACE-specific advantage. PACE's true differentiator at this operating point is instead *accuracy*: at the same 71.7 MB / 6.2 GB cost it attains a 65.6 five-benchmark average, $+1.0$ over SparseVLM and $+2.9$ over PyramidDrop, and slightly above the full model's 65.4. Second, the per-batch decode-latency figures should be read as indicative rather than exact: under batch > 1 serving they are sensitive to padding and KV-cache fragmentation across heterogeneous sequence lengths, which can make measured

Table 20: **Matched-retention batched-decode efficiency on Ours-2B.** At 25% retention SparseVLM / PyramidDrop and PACE keep the same 256 visual tokens, so their KV cache, GPU memory and batch-scaling envelope are essentially identical; the full model runs out of memory (OOM) at batch 64. Speedup / reduction factors in parentheses are relative to the full model at the same batch.

Batch	Method	Ret.	Decode (ms/tok)	KV Cache (MB)	GPU Mem (GB)
1	Ours-2B (Full)	100%	77.4	251.7	7.1
	SparseVLM	25%	76.9	71.7	6.0
	PyramidDrop	25%	76.6	71.7	6.0
	PACE	**25%**	**77.3**	71.7	**6.2**
8	Ours-2B (Full)	100%	83.6	2013.8	18.7
	SparseVLM	25%	82.5	573.8	11.9
	PyramidDrop	25%	83.3	573.8	11.9
	PACE	**25%**	**69.7**	575.4	**11.7**
16	Ours-2B (Full)	100%	81.5	4027.5	31.9
	SparseVLM	25%	80.5	1147.5	19.4
	PyramidDrop	25%	77.1	1147.5	19.4
	PACE	**25%**	**57.9**	1118.8	**17.9**
32	Ours-2B (Full)	100%	98.8	8055.0	58.5
	SparseVLM	25%	82.6	2295.0	32.4
	PyramidDrop	25%	83.7	2295.0	32.4
	PACE	**25%**	**58.3**	2237.5	**30.4**
48	Ours-2B (Full)	100%	135.4	12082.5	85.0
	SparseVLM	25%	81.3	3442.5	44.3
	PyramidDrop	25%	85.7	3442.5	44.3
	PACE	**25%**	**70.4** ($\times$**1.9**)	**3310.3** ($\times$**3.7**)	**43.0** ($\times$**2.0**)
64	Ours-2B (Full)	100%	OOM	OOM	OOM
	SparseVLM	25%	84.8	4590.0	59.3
	PyramidDrop	25%	86.1	4590.0	59.3
	PACE	**25%**	**85.6**	4430.7	**55.5**

latency non-monotonic in batch size (e.g., PACE's 57.9 ms/tok at batch 16). The consistent takeaway is that, at matched serving cost, PACE matches or exceeds the accuracy of both the full model and same-retention baselines.

G Limitations and Future Work

Limitations. Despite the consistent gains demonstrated above, PACE has three limitations that we explicitly acknowledge. *(i) Model scale.* Our experiments are conducted on a single 2B-parameter native MLLM built upon Qwen3-1.7B, and we do not provide a scaling study at 7B+ parameters. This is largely a practical constraint: to the best of our knowledge, no concurrent native MLLM fully open-sources all four training stages (PT / MT / SFT / RL) together with the required data recipes, which prevents a fair scaling comparison with prohibitive self-training cost. *(ii) Modality coverage.* Our evaluation focuses on image-text benchmarks. Video and audio tokens, where temporal redundancy is arguably more pronounced, are not covered by the current study. *(iii) Architecture dependence.* The full PACE method relies on multi-axis (T/H/W) Native-RoPE to compute directional SDIE scores. On backbones that expose such decoupled positional axes (*e.g.*, SAIL-7B), full PACE applies directly; on backbones with a single 1-D positional signal, only the PACS scheduling component is applicable, with SDIE degenerating to a scalar mean-attention criterion (the ablation in Table 1, Setting 2). We therefore stress that the ONECAT-3B setting is a scalar-scored PACS-only variant rather than a validation of full PACE on a genuine 1-D architecture, and that PACE's directional benefit is contingent on axis-decoupled position encoding.

Future Work. First, we will *scale PACE to 7B+ native MLLMs* and examine how model depth, width, and training stage shift Pareto optimal retention schedules and the relative benefits of SDIE and PACS. Second, we will *extend PACE to video and audio* by modeling temporal persistence and cross modal redundancy, testing whether axis aware scoring can compress long sequences without losing event level evidence. We will also study generalized directional scores for backbones without decoupled T/H/W positional axes, clarifying when SDIE should yield to PACS only scheduling.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction explicitly state the three main contributions of PACE: (i) a pilot study revealing that ViT-free native MLLMs tolerate aggressive compression while ViT-based MLLMs degrade; (ii) two probing findings on depth-progressive redundancy and cross-dimensional heterogeneity (Findings 1–2, §1); (iii) the PACE framework composed of SDIE (§2.2) and PACS (§2.3) that delivers near-lossless or improved performance at 25%–50% visual token retention with over 2× FLOPs reduction. These claims are directly supported by the experiments in §3 (Tables 1–5), the ablations in §3.4, the theoretical justifications in §B, and the qualitative cases in §E. The scope of claims is restricted to native MLLMs with decoupled multi-axis positional encoding (native M-RoPE), as explicitly stated in §1 and §2.1.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations in § G. Specifically, *(i) Model scale.* Our experiments are conducted on a single 2B-parameter native MLLM, and we do not provide a scaling study at 7B+ models. This is largely a practical constraint, that is prohibitive self-training cost. *(ii) Modality coverage.* Our evaluation focuses on image-text benchmarks. Video and audio tokens, where temporal redundancy is arguably more pronounced, are not covered by the current study.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.

- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: All theoretical results are stated formally with assumptions and complete proofs. Specifically, Theorem B.1 (Optimality of Additive-Multiplicative Fusion) in § B.1 justifies the SDIE fusion form, with the full proof proceeding in three steps (spatial additive composition, temporal multiplicative gating, monotonicity verification) under explicitly stated assumptions (separable bilinear function class, non-negativity, linearity in the temporal term). Theorem B.2 and the Pareto dominance analysis for PACS are stated in § B with proofs in the same appendix. All theorems and lemmas are numbered, cross-referenced in the main text, and the assumptions are explicitly listed in the theorem statements.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper fully discloses the information needed to reproduce the main experimental results. Specifically: (i) the native backbone is built on the public Qwen3-1.7B with architecture details (L_{buf}, L_{post}, hidden dim, head dim, native M-RoPE base frequencies, patch size, attention mask) fully specified in Table 11 (§ A.4); (ii) all four training stages (PT/MT/SFT/RL) are documented with data composition (Table 6–9, § A.2), optimizer, LR schedule, peak LR, warmup, global batch size, training steps, resolution, and precision in Table 10 (§ A.3); (iii) inference-time decoding settings (`greedy`, `temperature=0`, `topK=1`), evaluation tool-chain (VLMEvalKit with default prompts and `gpt-4o-1120` as judge), and hardware (H20-96G) are given in § A.5 and § 3.5; (iv) SDIE and PACS algorithms are formalized in § 2.2–2.3 with full Pareto search protocol in § D. Together these suffice to replicate PACE on the same Qwen3-1.7B backbone.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.

- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The PACE codebase is currently not publicly available due to company restrictions.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: All training and evaluation details are specified. § 3.1 gives the implementation summary (Qwen3-1.7B backbone, L_{buf}=4, M-RoPE base frequencies, 5×8 H20-96G GPUs, bfloat16). Full hyperparameters across the four training stages, including resolution schedule, optimizer (AdamW with β_1=0.9, β_2=0.999), weight decay 0.01, cosine LR schedule, peak LR, warmup, training steps, max sequence length, and global batch size, are listed in Table 10 (§ A.3). Architecture configuration is in Table 11 (§ A.4). Evaluation protocol (VLMEvalKit defaults, gpt-4o-1120 judge, greedy decoding, MME sum of perception+reasoning, HallusionBench averaged aAcc/qAcc/fAcc) is documented in § A.5 and the table captions of § 3.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We run PACS with 3 independently sampled calibration sets (different seeds, 1,000 images each) and report mean ± std on a representative subset of 3 benchmarks covering distinct capability domains (MMB, AI2D and HallBench) across all three retention levels. Full results are in Appendix C.2. Observed std values are consistently smaller than PACE-vs-baseline gaps, confirming statistical significance.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Compute resources are fully disclosed. § 3.1 and § A.3 state that all training stages use 5×8 H20-96G GPUs with bfloat16 precision; Table 10 gives per-stage training steps (PT: 250K, MT: 46K, SFT: 19K, RL: 2K) and global batch sizes (2560 / 1920 / 640 / 128 respectively). Inference efficiency experiments (§3.5) are run on a single H20-96G GPU at batch size 1 and 1024^2 input resolution, with per-stage latency, KV-cache, FLOPs, and GPU memory reported in Table 5. PACS's Pareto-front traversal takes ∼10 GPU-minutes on the same hardware (§ D.1). Preliminary and ablation experiments not reported in the main tables used the same cluster but significantly less compute; we acknowledge this in § A.3.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: We have reviewed the NeurIPS Code of Ethics and confirm that the research conforms in every respect. The work does not involve human subjects, crowd-sourcing, or personally identifiable information. All training data are from publicly released academic datasets (listed in Tables 6 – 9) used under their original licenses; we do not scrape new data. Model evaluation uses publicly available benchmarks. Anonymity is preserved in the current submission.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: Positive impacts: PACE reduces FLOPs by ∼2× and KV-cache by ∼3.5× on native MLLMs (§ 3.5) without sacrificing—and occasionally improving—benchmark accuracy, making multimodal inference more accessible on resource-constrained devices and reducing the energy cost of large-model serving. It also accelerates the adoption of the native/ViT-free MLLM paradigm, which removes the brittle modality-boundary assumption of modular MLLMs. Potential negative impacts: as a general efficiency technique, PACE inherits the societal risks of any performant MLLM, including potential misuse for automated disinformation, non-consensual surveillance, or biased decision-making. PACE itself neither amplifies nor mitigates these risks beyond the underlying backbone. We do not introduce a new generative model with direct deployment applications beyond the already-public Qwen3-1.7B-based backbone.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: PACE is a visual-token compression framework applied to an existing public backbone. It does not introduce new generative capabilities, does not scrape or release new datasets, and does not increase the risk profile of the underlying model beyond what is already released in the public domain. Consequently, no additional safeguards beyond those already provided by the underlying backbone are required. Upon code release, we will follow standard licensing and usage-guideline practices aligned with Qwen3's license.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All existing assets are properly credited. The Qwen3 backbone [6; 106] is used under its original Apache-2.0 license. Training data sources, including all datasets in Table 6– 9, are each cited to their original publications with their respective licenses respected. Evaluation benchmarks (MMB, AI2D, MME, HalBench, SEED, RWQA, MMStar, MathVista) are cited in Table 1– 5. The VLMEvalKit [31] tool-chain is used under its public license.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.

- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: The PACE codebase, model checkpoint, and accompanying documentation are currently not publicly available due to company restrictions.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowd-sourcing or research with human subjects. All datasets used are publicly available academic benchmarks. No new human annotations are collected in this work.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve human-subjects research, so IRB approval is not applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [No]

Justification: LLMs are not used as an original, non-standard component of the core methodology.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.