# VLZip: Unified Visual and Textual Compression for Interleaved Long-Context Modeling

Yuqi Zhang[1,2,3⋆], Cheng Chen[3⋆], Yuyu Guo[3], Wenjie Yang[3], Lingchen Meng[1], Peng Di[3], Hang Yu[3](✉), Zuxuan Wu[1,2](✉), and Yu-Gang Jiang[1](✉)

[1] Shanghai Key Lab of Intell. Info. Processing, School of CS, Fudan University, China
[2] Shanghai Innovation Institute, China
[3] Ant Group, China
zxwu@fudan.edu.cn, hyu1@e.ntu.edu.sg

**Abstract.** Vision Language Models (VLMs) face significant challenges with ultra-long, interleaved image-text sequences due to the quadratic complexity of self-attention. Current solutions either resort to aggressive token pruning, risking irreversible information loss, or adopt efficient but less precise architectures, while largely ignoring the equally vital textual component. We introduce VLZip, a framework that unifies visual and textual compression for high-fidelity reasoning within a pure Transformer. At its core, VLZip hierarchically distills visual and textual segments into compact, layer-specific "soft prefixes" and injects them into each decoder layer's hidden states, drastically shortening the attention sequence while preserving fine-grained global context. To address deficient evaluations in the field, we also introduce LongVLBench, a new benchmark derived from video narratives that demands holistic, narrative-level reasoning. Extensive experiments show VLZip achieves leading performance on long-context multimodal reasoning, enabling training up to 120K tokens—a 6× increase over the baseline—and inference beyond 280K tokens with significantly reduced memory, while demonstrating the memory scalability to handle up to 2M tokens. By excelling at extreme context lengths where existing methods collapse, VLZip establishes an efficient and powerful new standard for long-context multimodal AI. Code is available at https://github.com/ShareLab-SII/VLZip.

**Keywords:** Long-Context Vision-Language Models · Multimodal Token Compression · Interleaved Image-Text Reasoning

## 1 Introduction

The celebrated success of VLMs [1, 13, 16, 17] has been largely confined to a world of snapshots, not narratives. While highly proficient at processing single

⋆ Equal contribution. This work was done when Yuqi Zhang was a research intern at Ant Group.

images, they struggle with the ultra-long, interleaved sequences of images and text that define complex real-world activities. These applications span a wide spectrum—from following detailed illustrated manuals and analyzing a patient's comprehensive visual medical history, to evaluating long-form video narratives and powering multimodal GUI agents that autonomously operate over extended horizons. Across all these domains, the next frontier for AI is not merely to process more tokens, but to unlock a new class of **long-context intelligence** capable of holistic, narrative-level understanding.

However, this critical domain remains largely underexplored. Progress is stalled by two fundamental, interconnected hurdles: the first is **architectural**, with the quadratic scaling of self-attention [27] making long sequences computationally prohibitive; the second is **evaluative**: the benchmarks used to measure progress are poorly suited to narrative reasoning[1], which steers the research community away from solving the real problem.

On the architectural side, existing approaches typically address this challenge by accepting a trade-off between efficiency and fidelity (as shown in Fig. 1). One path involves aggressive input pruning [35], a strategy that irreversibly discards tokens—almost exclusively from the visual modality—risking significant information loss. A second path attempts an architectural overhaul, replacing the Transformer with more efficient but potentially less expressive backbones like State Space Models [29, 33], which can compromise the precise reasoning at which Transformers excel. A third, data-centric strategy [9] improves reasoning through high-quality data but fails to address the computational bottleneck. Yet none offers a principled path to simultaneously high efficiency and high-fidelity reasoning within a pure Transformer.

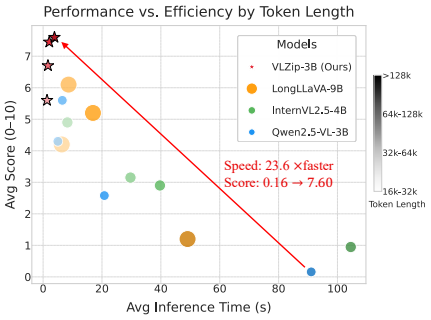



**Fig. 1:** A comparison of performance and efficiency for VLZip(Ours, red star) and baseline models. Performance across context lengths is indicated by the color intensity of the points.

On the evaluation side, the few existing long-context multimodal benchmarks are equally unsatisfying. Some, such as Mantis-Eval [9] and MileBench [26], are built by repurposing short-context datasets, leading to imbalanced image-to-text ratios or insufficient context lengths. Others rely on artificial "needle-in-a-haystack" (NIAH) retrieval tasks, as seen in MM-NIAH [28] and MM-LongCite [37], which test for simple fact recall rather than holistic comprehension. MMLongBench [30], for instance, creates length by padding with irrelevant

[1] Throughout this paper, we use the term *reasoning* to refer to the model's broad capacity for interpreting and synthesizing multimodal information, as distinct from explicit chain-of-thought or deliberative "thinking" paradigms.

filler documents, introducing noise and risking data contamination. This creates a self-perpetuating limitation: models are not designed for genuine narrative intelligence because the benchmarks do not demand it.

We argue that true progress requires a unified solution that addresses both modeling and evaluation. In this paper, we introduce VLZip, a framework that redefines the efficiency-fidelity curve by introducing a more intelligent trade-off. Instead of sacrificing reasoning or irreversibly discarding information, VLZip trades a small, one-time compression cost for massive, sustained efficiency gains within a pure Transformer architecture. At its core, VLZip operates on a principle of hierarchical context distillation and multi-layer injection. It first distills extensive image and text segments into compact, information-rich "soft prefixes", and then injects these prefixes into the hidden states of every decoder layer. This provides the model with a constant, fine-grained stream of global context, allowing it to maintain a high-fidelity understanding of the entire narrative while operating on a drastically shortened sequence. As a preview of our results, Fig. 1 illustrates how VLZip substantially improves the existing performance-efficiency trade-off, achieving state-of-the-art accuracy while being over an order of magnitude faster than baselines on ultra-long contexts.

To properly validate VLZip and address the field's evaluation gap, we also construct a new, challenging benchmark, **LongVLBench**, derived from video narratives. This benchmark provides long, chronologically coherent sequences that force models to move beyond simple retrieval and engage in authentic narrative-level reasoning. Our extensive experiments demonstrate that VLZip achieves strong performance on MMLongBench—particularly excelling on in-context learning tasks at extreme lengths—and sets a new state-of-the-art on our narrative-focused LongVLBench where other methods falter. It enables training on sequences up to 120K tokens on standard hardware—a 6× improvement over the baseline—while drastically reducing computational and memory costs.

In summary, the contributions of this paper are:

- We introduce VLZip, a unified visual-textual compression framework that enables training on sequences up to 120K tokens and inference beyond 280K tokens, while demonstrating a memory-efficient design that shows a clear path to scaling to 2M tokens within a pure Transformer architecture.
- We propose hierarchical context distillation with multi-layer injection that, for the first time, unifies the compression of both visual and textual modalities into layer-specific "soft prefixes" injected into every decoder layer.
- We construct a benchmark, LongVLBench, for evaluating long-context narrative understanding. Derived from video, it provides chronologically coherent interleaved sequences that address the shortcomings of existing benchmarks (e.g., artificial tasks, imbalanced contexts) and require narrative reasoning.
- We empirically demonstrate that VLZip both sets a new state-of-the-art (SOTA) on long-context multimodal reasoning benchmarks and enables training on sequences up to 120K tokens on standard hardware—a 6-fold improvement over the baseline—establishing a practical standard for long-context AI.

## 2 Related Works

The pursuit of long-context VLMs is an emerging field, with progress confronting challenges in both model architecture and evaluation. A detailed review is provided in the supplementary material; here, we contextualize our work by briefly surveying the dominant strategies.

**Architectures.** Current long-context VLM architectures have largely centered on three strategies, each forcing a compromise between performance and efficiency. The first, Input Pruning [32, 35], permanently discards visual tokens to shorten sequences, risking irreversible information loss while ignoring the equally vast textual information present in interleaved multimodal sequences. A second path, Architectural Overhaul [29,33], replaces the Transformer's attention mechanism with more scalable backbones like State Space Models, but can sacrifice the precise, non-local reasoning capabilities at which Transformers excel. A third, Data-Centric approach [9] enhances multi-image reasoning with high-quality data but does not resolve the computational bottleneck for long sequences. Our work is inspired by a promising alternative from the NLP domain: Soft Prompt Compression [5–7, 15]. Methods such as GIST [23] and the In-Context Autoencoder [7] demonstrate that long textual contexts can be effectively distilled into compact, information-rich soft prompts, offering efficiency without aggressive pruning. However, their application has been confined to the text modality and typically involves a simple, single-layer injection. Concurrently, DeepStack [22] explores multi-layer stacking for visual tokens but does not address textual compression or interleaved multimodal contexts. VLZip bridges these two lines of work by unifying both visual and textual compression with multi-layer injection in a single framework, preserving high-fidelity context throughout decoding.

**Evaluations.** Paralleling the architectural challenges, existing multimodal long-context benchmarks also exhibit shortcomings. Many are constructed by repurposing short-context datasets [9, 26], relying on artificial 'needle-in-a-haystack' retrieval tasks [28, 37], or padding with irrelevant filler documents [30]. These limitations motivate our development of LongVLBench, a benchmark built from the ground up with chronologically coherent, narrative-driven sequences to enable a more authentic evaluation of holistic reasoning.

## 3 Method

We introduce VLZip, a unified compression framework for efficient, high-fidelity reasoning over long interleaved image-text sequences. A complete input sequence, formally expressed as $\mathcal{X} = \{\mathcal{S}, \mathcal{C}, \mathcal{Q}\}$, comprises a system prompt $\mathcal{S}$, a question $\mathcal{Q}$, and the lengthy interleaved context $\mathcal{C}$. To reduce this cost, VLZip compresses only the context $\mathcal{C}$, leaving the prompt and question untouched. Instead of feeding the full context to the VLM, it represents each image and text segment with compact placeholder tokens, drastically shortening the effective input length.

The interleaved context $\mathcal{C} = \{t_0, \mathbf{v}_0, \ldots, t_{n-1}, \mathbf{v}_{n-1}, t_n\}$, where $t_i$ are text segments and $\mathbf{v}_j$ are images, is processed externally by modality-specific compressors. These hierarchical compressors distill each image and text segment

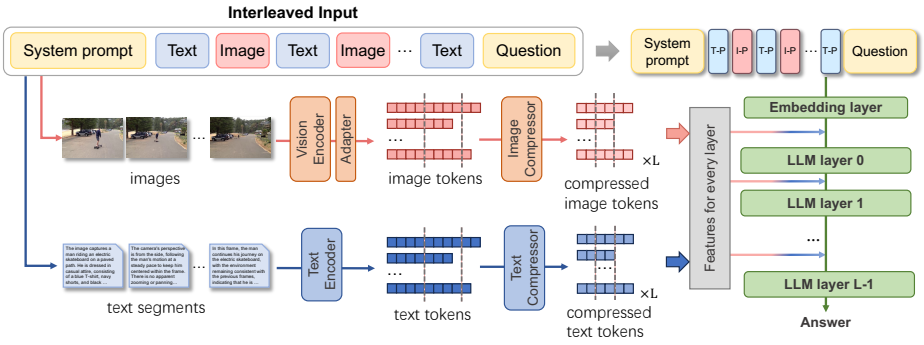



**Fig. 2: Overview of VLZip Architecture.** VLZip compresses interleaved image-text sequences via modality-specific hierarchical modules. The visual module partitions each image into chunks and compresses each into $M_v$ layer-specific tokens via a shared Q-Former. The textual module partitions each text segment into chunks, encodes each chunk with a lightweight text encoder, and compresses each into $M_t$ layer-specific tokens via a shared Q-Former. Compressed features are injected into each LLM decoder layer via element-wise addition. T-P and I-P denote text and image placeholders.

into compact, layer-specific soft-prefix features. Rather than being appended to the input as additional tokens, these features are injected at the positions held by placeholder tokens during decoding. Critically, at each decoder layer, the corresponding compressed features are injected into the hidden states of the placeholders, ensuring continuous, fine-grained access to global context and overcoming information loss from single-step pruning or shallow injection.

### 3.1 Architecture

As illustrated in Figure 2, the VLZip framework is built upon a standard VLM backbone and comprises three key components: (1) a **Hierarchical Visual Compressor** that encodes each image into a set of layer-specific feature vectors; (2) a **Hierarchical Textual Compressor** that similarly distills long text segments into layer-specific features; and (3) an **Information Injection Mechanism** that injects the compressed features into dedicated placeholder tokens at every decoder layer. The placeholder tokens preserve the compact sequence structure, while the layer-specific soft-prefix features carry the semantic information of the original context.

**Hierarchical Visual Compression Module.** In a conventional VLM, each image $v_j$ is passed through a vision encoder and a projection adapter to produce a sequence of visual tokens $\mathbf{F}_{v,j} \in \mathbb{R}^{N_{v,j} \times d}$, where $d$ is the VLM's hidden dimension and the token count $N_{v,j}$ can be hundreds or thousands. To avoid this bottleneck, we adaptively compress these $N_{v,j}$ tokens into a smaller set of $N'_{v,j}$ tokens.

Specifically, each image is uniformly partitioned into chunks of size $C_v$, with each independently processed by a shared image compressor that generates $M_v$

compressed visual tokens per decoder layer. The total number of compressed tokens per image is thus $N'_{v,j} = \lceil N_{v,j}/C_v \rceil \times M_v$.

To avoid irreversible information loss, the image compressor employs a hierarchical strategy by producing layer-specific compressed features for all $L$ decoder layers simultaneously. It utilizes a Q-Former architecture with a shared set of $L \times M_v$ learnable queries $\mathbf{q}_v \in \mathbb{R}^{(L \cdot M_v) \times d}$, allowing all layer-specific tokens to be generated in a single forward pass. For a given image chunk $\mathbf{F}_{v,j,p} \in \mathbb{R}^{C_v \times d}$ (the $p$-th chunk of image $v_j$), the compression process is:

$$\mathbf{Q}_v = \mathrm{LN}(\mathbf{q}_v); \quad \mathbf{K}_v = \mathrm{LN}(\mathbf{F}_{v,j,p} + \mathrm{PE}(\mathbf{F}_{v,j,p})); \quad \mathbf{V}_v = \mathrm{LN}(\mathbf{F}_{v,j,p}) \tag{1}$$

$$\mathbf{Z}_{v,j,p} = \mathrm{MHA}(\mathbf{Q}_v, \mathbf{K}_v, \mathbf{V}_v) \tag{2}$$

Here, $\mathrm{LN}(\cdot)$ denotes Layer Normalization, $\mathrm{PE}(\cdot)$ denotes Positional Encoding, and $\mathrm{MHA}(\cdot)$ denotes Multi-Head Attention. The output $\mathbf{Z}_{v,j,p} \in \mathbb{R}^{(L \cdot M_v) \times d}$ is reshaped into $\mathbb{R}^{L \times M_v \times d}$, where the $l$-th slice $\mathbf{Z}^{(l)}_{v,j,p} \in \mathbb{R}^{M_v \times d}$ gives the $M_v$ compressed visual tokens for chunk $p$ tailored for decoder layer $l$.

**Hierarchical Text Compression Module.** Long text segments present a similar computational burden. For a given text segment $t_i$ with $N_{t,i}$ tokens, we first uniformly partition it into $K_i = \lceil N_{t,i}/C_t \rceil$ chunks of size $C_t$. Each chunk is then independently encoded by a lightweight text encoder $E(\cdot)$ to obtain chunk-level hidden states $\mathbf{F}_{t,i,k} = E(t_{i,k}) \in \mathbb{R}^{C_t \times d_e}$. Each chunk is then processed by the text compressor, which projects it to the VLM's hidden dimension $d$ and compresses it into $M_t$ tokens for each of the $L$ decoder layers. The text compressor utilizes a Q-Former architecture with a shared set of $L \times M_t$ learnable queries $\mathbf{q}_t \in \mathbb{R}^{(L \cdot M_t) \times d}$ to generate all layer-specific tokens in a single forward pass. The compression of a single chunk $\mathbf{F}_{t,i,k}$ is:

$$\mathbf{F}'_{t,i,k} = \mathbf{F}_{t,i,k}\mathbf{W}_{\text{proj}} \tag{3}$$

$$\mathbf{Q}_t = \mathrm{LN}(\mathbf{q}_t); \quad \mathbf{K}_t = \mathrm{LN}(\mathbf{F}'_{t,i,k} + \mathrm{PE}(\mathbf{F}'_{t,i,k})); \quad \mathbf{V}_t = \mathrm{LN}(\mathbf{F}'_{t,i,k}) \tag{4}$$

$$\mathbf{Z}_{t,i,k} = \mathrm{MHA}(\mathbf{Q}_t, \mathbf{K}_t, \mathbf{V}_t) \tag{5}$$

where $\mathbf{W}_{\text{proj}} \in \mathbb{R}^{d_e \times d}$ is a projection matrix mapping from the text encoder's hidden dimension $d_e$ to the VLM's hidden dimension $d$. The output $\mathbf{Z}_{t,i,k} \in \mathbb{R}^{(L \cdot M_t) \times d}$ is reshaped into $\mathbb{R}^{L \times M_t \times d}$, where the $l$-th slice $\mathbf{Z}^{(l)}_{t,i,k} \in \mathbb{R}^{M_t \times d}$ gives the $M_t$ compressed text tokens for chunk $k$ tailored for decoder layer $l$.

**Information Injection.** The VLM decoder operates on a compact sequence consisting of the system prompt $\mathcal{S}$, the question $\mathcal{Q}$, and a placeholder sequence $\mathcal{P}$ representing the compressed image and text segments. For each image chunk $\mathbf{F}_{v,j,p}$, $\mathcal{P}$ contains $M_v$ visual placeholders, while for each text chunk $\mathbf{F}_{t,i,k}$, it contains $M_t$ textual placeholders. At decoder layer $l$, the hidden states of these placeholders are augmented with the corresponding compressed features before self-attention. Let $\mathbf{H}^{(l)}_{v,j,p} \in \mathbb{R}^{M_v \times d}$ and $\mathbf{H}^{(l)}_{t,i,k} \in \mathbb{R}^{M_t \times d}$ be the hidden states of the placeholders for the $p$-th chunk of image $v_j$ and the $k$-th chunk of text segment $t_i$, respectively. The injection is:

$$\hat{\mathbf{H}}^{(l)}_{v,j,p} = \mathbf{H}^{(l)}_{v,j,p} + \mathbf{Z}^{(l)}_{v,j,p}, \quad \hat{\mathbf{H}}^{(l)}_{t,i,k} = \mathbf{H}^{(l)}_{t,i,k} + \mathbf{Z}^{(l)}_{t,i,k} \tag{6}$$

This element-wise addition fuses the distilled global context directly into the model's processing pipeline at every layer. The full sequence of updated hidden states is then fed into the layer's self-attention module. This allows the VLM to reason with a complete understanding of the long context while the attention mechanism operates only on a short, computationally feasible sequence.

### 3.2 Training Pipeline

To effectively train the various components of VLZip, we design a four-stage progressive training pipeline. This curriculum enables the model to first master modality-specific compression and then learn to integrate these compressed representations for complex, long-context reasoning.

**Stage 1: Visual Compressor Pre-training.** We train the visual compressor on large-scale single-image instruction data (e.g., LLaVA-OneVision [11] single-image data). Only the projection adapter and vision Q-Former parameters are trainable. This stage teaches the compressor to distill visual tokens into compact, layer-specific representations $\mathbf{Z}_{v,j}^{(l)}$.

**Stage 2: Textual Compressor Pre-training.** We pre-train the textual compressor using a reconstruction task on text-only long-document corpora (e.g., ChatQA2 [31]). The model learns to compress each text chunk into tokens $\mathbf{Z}_{t,i,k}$ and reconstruct the original text. Only the text compressor parameters (the lightweight encoder, $\mathbf{W}_{\text{proj}}$, and text Q-Former) are trained. This self-supervised objective ensures the compressed tokens preserve semantic completeness.

**Stage 3: Joint Interleaved Fine-tuning.** With both compressors pre-trained, this stage aims to teach the VLM how to fuse and reason over compressed information from both modalities simultaneously. We use datasets containing interleaved sequences of multiple images and texts. In this stage, we keep the compressors trainable and unfreeze the LLM decoder. This step is crucial for harmonizing the independently trained modules within the unified architecture.

**Stage 4: Long-Context Consolidation.** Building on previous stages, we continue training on long-form textual data with all Stage 3 components remaining trainable. This final stage adapts the model's reasoning strategies to ultra-long contexts, ensuring robust performance across extensive image-text sequences. Complete hyperparameters are provided in the supplementary material.

## 4 LongVLBench: A Narrative-Driven Benchmark for Long-Context Evaluation

As discussed in Sections 1 and 2, existing long-context multimodal benchmarks are limited by several notable shortcomings. They often repurpose disconnected, short-context data, rely on artificial "needle-in-a-haystack" (NIAH) retrieval tasks that test recall over reasoning, or pad sequences with irrelevant filler documents. These shortcomings create a landscape where models are not evaluated on—and therefore not optimized for—genuine narrative-level understanding. To address this critical evaluation gap, we introduce **LongVLBench**, a new

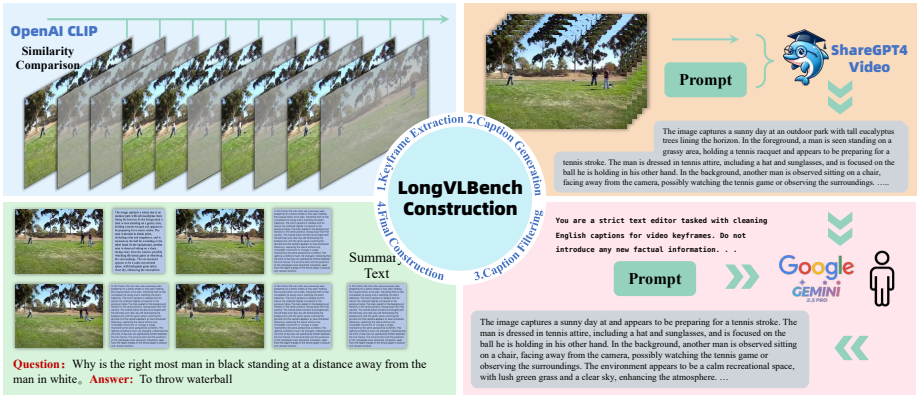



**Fig. 3: Overview of our four-stage dataset creation pipeline.** The process begins with (1) extracting semantic keyframes from videos using CLIP-based similarity comparison. Next, (2) ShareGPT4Video generates detailed, transitional, and summary captions for the frames. These captions are then (3) rigorously refined through a prompt-guided LLM and human review to eliminate redundancy and ensure factual fidelity. Finally, (4) the cleaned captions and their corresponding keyframes are assembled into the final long-context, interleaved dataset.

benchmark constructed from the ground up to provide a rigorous and authentic testbed for long-form multimodal intelligence. Sourced from video narratives, our systematic pipeline (Figure 3) is purpose-built to generate genuinely long, chronologically coherent, and densely interleaved sequences that demand holistic comprehension. The process consists of four key stages.

**Semantic Keyframe Extraction.** To keep the true visual continuity, our pipeline begins by converting video into a discrete sequence of keyframes. Unlike methods that stitch together unrelated images, we employ a semantically aware extraction algorithm using CLIP embeddings [24]. A new keyframe is selected only when a significant visual change is detected, ensuring the resulting image sequence captures the video's core narrative arc. This process yields a chronologically coherent visual story, overcoming the limitations of benchmarks built from repurposed, disjointed data.

**Hierarchical Narrative Captioning.** To create a context where text is an integral part of the story rather than just filler, we generate a rich, multi-layered textual narrative for the keyframes. Using a powerful video language model (ShareGPT4Video) [3], we produce three levels of description: First, it generates a detailed caption for each individual keyframe. Then, by providing context from previous frames, it creates transitional descriptions that highlight the progression between them. Finally, a comprehensive summary caption is generated for the entire keyframe sequence to capture the video's overarching theme.

**Rigorous Refinement for Factual Fidelity.** To ensure LongVLBench is a reliable benchmark, all auto-generated captions undergo stringent, hybrid refinement. We use a SOTA LLM (Gemini 2.5 Pro) guided by strict principles to

eliminate redundancy, normalize style, and enhance fluency. Most importantly, the process is explicitly constrained to prevent the model from hallucinating facts or inferring details not present in the source material. Human review provides a final layer of quality control, guaranteeing that the final textual narrative is coherent, concise, and factually grounded in the visual evidence.

**Final Interleaved Data Construction.** One crucial step is designing tasks that measure holistic reasoning, not simple fact retrieval. For each long-form document, we generate QA pairs that explicitly require synthesizing information across long spans of interleaved text and images. Questions are designed to be unanswerable by observing a single frame or reading a single caption, compelling the model to understand plot progression, character interactions, and evolving context. The refined, interleaved narratives and corresponding QA pairs are assembled into final documents. From this collection, we constructed a high-quality test set of 140 samples, carefully selected to represent a diverse distribution of token lengths (Fig. 4) and provide a challenging test for ultra-long multimodal reasoning.

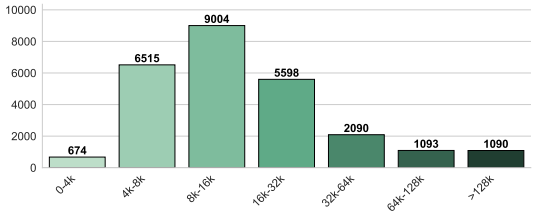



**Fig. 4:** Token length distribution of the interleaved dataset. The histogram shows the frequency of the 26,164 generated documents across seven token length bins.

## 5 Experiments

### 5.1 Experiment Setup

**Implementation Details.** Our model uses the Qwen2.5-VL-Instruct-3B backbone [2]. Both image and text inputs are segmented into chunks of 100 tokens, each compressed into 4 tokens per decoder layer via modality-specific Q-Former compressors. Text chunks are encoded by Qwen2.5-0.5B prior to compression. The model is trained over four stages on 32×A100 or 32×H20 GPUs. Additional hyperparameters are detailed in the supplementary material.

**Evaluation Benchmarks.** We assess long-context understanding using our proposed **LongVLBench** and the established **MMLongBench** [30], featuring input lengths from 8K to 128K tokens. To assess foundational abilities, we also test on several short-context VQA benchmarks via the **LMMs-Eval** framework [36].

**Baselines.** We compare our approach against models with distinct architectural strategies: dedicated long-context models (LongLLaVA [29], Long-VITA [25]), interleaved reasoning models (Mantis [9], InternVL2.5 [4], Ovis2 [20]), and other compression methods (GlimpsePrune [35], VisionZip [32]). For a fair comparison, the compression-based methods are implemented on the same Qwen2.5-VL-3B backbone. The unmodified base model is used as a reference.

**Table 1:** LongVLBench performance across different context lengths. Colored numbers indicate the inference success rate (100% denotes full success, lower rates indicate OOM or no responses). The overall average is computed as the total score divided by the total number of samples across all length ranges and scaled to 0-100 range. **Bold** and underline indicate the best and second best performance respectively.

| Model | Size | 0-4k | 4k-8k | 8k-16k | 16k-32k | 32k-64k | 64k-128k | >128k | Avg |
|---|---|---|---|---|---|---|---|---|---|
| Long-VITA [25] | 14B | 5.25 100% | 6.10 100% | **6.94** 80% | 3.80 75% | 5.78 45% | 5.33 15% | 0.00 0% | 33.1 |
| LongLLaVA [29] | 9B | 4.70 100% | 5.15 100% | 4.95 100% | 4.20 100% | 6.10 100% | 5.20 100% | 1.20 100% | 45.0 |
| Mantis [9] | 8B | 0.70 100% | 2.00 100% | 1.88 85% | 0.00 10% | 2.00 5% | 0.00 0% | 0.00 0% | 6.30 |
| Qwen2.5-VL [2] | 7B | 4.90 100% | 5.55 100% | 5.80 100% | 5.25 100% | 5.70 100% | 4.25 100% | 1.58 95% | 47.1 |
| InternVL2.5 [4] | 4B | 4.95 100% | 5.05 100% | 5.85 100% | 4.90 100% | 3.15 100% | 2.90 100% | 0.95 95% | 39.1 |
| Ovis2 [20] | 4B | 4.85 100% | **6.25** 100% | 6.25 100% | 4.35 100% | 5.95 100% | 0.4 100% | 0.5 100% | 40.8 |
| GlimpsePrune [35] | 3B | **5.35** 100% | 4.70 100% | 6.20 100% | 4.45 100% | 5.10 100% | 2.21 95% | 0.00 65% | 39.9 |
| VisionZip [32] | 3B | 4.95 100% | 4.70 100% | 6.89 90% | 3.75 20% | 2.00 5% | 4.00 15% | 0.00 0% | 24.7 |
| Qwen2.5-VL [2] | 3B | **5.35** 100% | 4.90 100% | 6.20 100% | 4.30 100% | 5.60 100% | 2.58 95% | 0.16 95% | 41.4 |
| VLZip (Ours) | 3B | 4.60 100% | 5.65 100% | 5.75 100% | **5.60** 100% | **6.70** 100% | **7.45** 100% | **7.60** 100% | **61.9** |

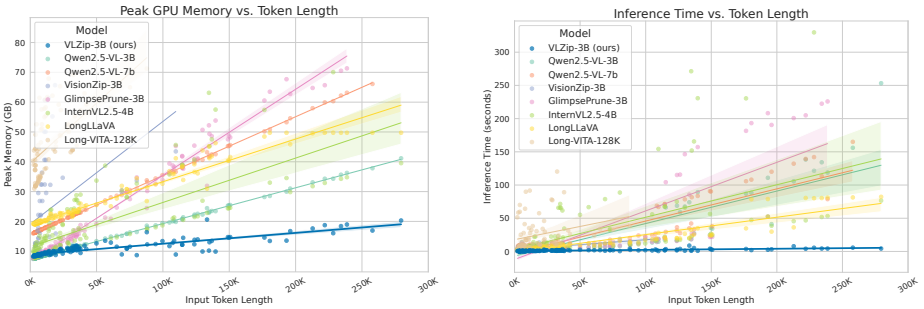
### 5.2 Main Results

**Results on LongVLBench.** On our challenging LongVLBench, designed to test authentic narrative-level reasoning, VLZip sets a new state-of-the-art with an average score of **61.9**, a 31.4% relative improvement over the next-best model (Tab. 1). This success stems from our unified compression, which enables robust processing of all test samples at 100% inference success rate across all length ranges. Despite being 3× smaller, our 3B model consistently outperforms the 9B LongLLaVA. The advantage is starkest at extreme lengths: while the baseline Qwen2.5-VL-3B model nearly fails on inputs exceeding 128k tokens (scoring only 0.16), VLZip maintains strong performance at 7.60, an absolute improvement of 7.44 points, highlighting that unified compression is critical for sustaining narrative comprehension at ultra-long scales.

**Results on MMLongBench.** To further validate generalization, we evaluate on three MMLongBench tasks: visual retrieval-augmented generation (VRAG), needle-in-a-haystack (NIAH), and many-shot in-context learning (ICL). While showing a modest performance trade-off on shorter-context retrieval tasks compared to the baseline—a characteristic of its specialization, which we analyze in the sequel—Tab. 2 demonstrates that VLZip's advantages become increasingly pronounced at extreme context lengths. At 128k tokens, where most models degrade severely, VLZip achieves **23.3** on VRAG and **25.3** on NIAH, ranking first among all models of comparable scale (<7B parameters) and greatly outperforming other compression methods like GlimpsePrune and VisionZip. Most strikingly, VLZip achieves **58.8** on ICL-128k, substantially outperforming all competing models including Qwen2.5-VL-7B (44.0)—a model twice its size. This exceptional ICL performance directly validates our multi-layer injection

**Table 2:** Comparison on VRAG, NIAH, and ICL tasks of MMLongBench. *Text segments shorter than 100 characters are passed through without compression, as such short segments yield minimal efficiency gains while compression may discard critical information. **Bold** and underline indicate the best and second best.

| Metric | | Large Scale Models (>=7B) | | | Small Scale Models (<7B) | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Size (Params) | Long-VITA-128K 14B | LongLLAVA 9B | Qwen2.5-VL 7B | InternVL2.5 4B | Ovis2 4B | GlimpsePrune 3B | Qwen2.5-VL 3B | VisionZip 3B | VLZip* 3B |
| VRAG | 8k | 50.3 | 31.9 | 48.3 | 40.9 | 35.0 | 40.6 | **43.4** | 41.7 | 25.2 |
| | 16k | 50.9 | 31.3 | 45.3 | 38.6 | 30.7 | 37.4 | 38.1 | **38.7** | 28.1 |
| | 32k | 41.8 | 28.1 | 43.9 | 30.5 | 29.4 | 33.2 | 34.4 | **35.3** | 26.3 |
| | 64k | 46.5 | 27.1 | 35.1 | 30.9 | 31.7 | 28.7 | **34.6** | 33.6 | 27.0 |
| | 128k | - | 18.9 | 31.1 | 21.3 | 4.7 | 8.2 | 10.8 | 8.6 | **23.3** |
| NIAH | 8k | 63.1 | 48.7 | 56.1 | **59.0** | 49.0 | 56.2 | 55.3 | 52.3 | 35.7 |
| | 16k | - | 47.6 | 52.1 | **55.2** | 37.9 | 51.7 | 50.5 | - | 32.1 |
| | 32k | - | 42.4 | 45.6 | **49.2** | 29.5 | 44.2 | 43.3 | - | 30.8 |
| | 64k | - | 39.5 | 33.8 | **41.8** | 26.9 | 34.3 | 35.2 | - | 28.5 |
| | 128k | - | 36.5 | 27.0 | 25.4 | 13.2 | 13.5 | 13.4 | - | **25.3** |
| ICL | 8k | - | 76.5 | 97.5 | 93.1 | 64.8 | 90.6 | 93.1 | 91.2 | **94.1** |
| | 16k | - | 52.4 | 91.2 | 78.6 | 13.8 | 55.3 | 68.1 | - | **87.5** |
| | 32k | - | 39.3 | 81.5 | 51.0 | 5.5 | 16.5 | 20.0 | - | **79.8** |
| | 64k | - | 16.8 | 57.5 | 8.8 | 0.5 | 4.8 | 7.3 | - | **63.0** |
| | 128k | - | - | 44.0 | 0.8 | 0.5 | 6.8 | 7.5 | - | **58.8** |




**(a)** The peak GPU memory usage versus input token length on the LongVLBench dataset.

**(b)** The inference time versus input token length on the LongVLBench dataset.

**Fig. 5:** Analysis of resource consumption (memory and time) versus input token length on the LongVLBench dataset.

design: by continuously providing fine-grained global context to every decoder layer, VLZip retains the ability to leverage long-range in-context examples even at extreme lengths where other methods collapse.

**Inference Efficiency and Scalability.** Our architectural design translates directly into major computational gains. As shown in Figs. 5a and 5b, peak GPU memory scales significantly more favorably, enabling the processing of sequences over 280k tokens on a single A100-80GB GPU—a scale unattainable by baselines. Inference speed similarly benefits, with VLZip running 23.6× faster than the base model at extremely long context lengths. To further isolate memory scaling behavior, we measure prefilling peak memory on synthetic interleaved sequences with a 1:1 image-to-text token ratio across lengths from 32K to 2M tokens (Fig. 6). All baselines exhibit steep memory growth: Long-VITA exceeds the 80GB A100 limit before 128K tokens, GlimpsePrune surpasses it near 256K, and

**Table 3:** Performance on short-context VLM benchmarks: MMBench [18], GQA [8], POPE [14], AI2D [10], MMMU [34], SEED-Bench-Image ($SEED^{img}$) [12], ScienceQA-Image ($SQA^{img}$) [19], and OKVQA [21].

| Model | MMBench | GQA | POPE | AI2D | MMMU | $SEED^{img}$ | $SQA^{img}$ | OKVQA |
|---|---|---|---|---|---|---|---|---|
| Qwen2.5-VL-3B | 78.4 | 60.0 | 87.6 | 78.6 | 46.9 | 74.8 | 81.1 | 42.5 |
| VLZip-3B | 72.9 | 53.8 | 83.4 | 71.6 | 43.1 | 67.9 | 79.2 | 51.1 |

LongLLaVA reaches the limit around 512K tokens. The uncompressed baseline (Qwen2.5-VL) similarly runs out of memory beyond 256K tokens. In contrast, VLZip's memory curve flattens after 512K tokens by batch-processing images through the visual compressor before decoding, amortizing the visual encoding cost. This allows VLZip to scale to **2M** tokens—an order of magnitude beyond all baselines—while keeping peak memory under 50GB on a single card. Together, the minimal memory overhead and fast inference establish VLZip as a practical solution for long-context applications.

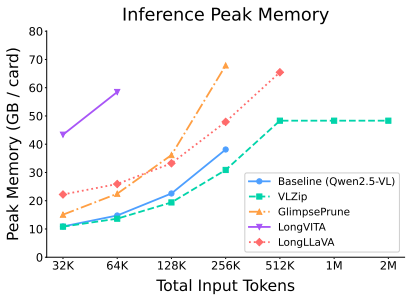



**Fig. 6:** Prefilling peak memory vs. input token length.

**Short-Context Performance.** To assess the impact of our specialization, we evaluated VLZip on standard short-context benchmarks (Tab. 3). Since these benchmarks involve only short textual queries, text compression is not activated, and only visual compression is applied. As anticipated, and consistent with the trade-offs observed on shorter-context MMLongBench tasks, the compression-focused design introduces a moderate trade-off on perception-heavy tasks such as GQA and AI2D, while performance remains competitive on knowledge-heavy VQA tasks, with VLZip even surpassing the baseline on OKVQA. Overall, VLZip retains over 90% of the baseline's performance across most benchmarks, confirming that long-context compression capability is achieved without fundamentally compromising core vision-language abilities.

### 5.3 Ablation Studies

Next, we conduct a series of ablation studies to dissect VLZip's core components and validate our key design choices.

**Necessity of Unified Compression.** We validated the necessity of our compression modules by measuring the maximum trainable sequence length on 8×A100 GPUs with DeepSpeed ZeRO-2, tested in steps of 10K tokens (Table 4). The uncompressed baseline model fails beyond 20K tokens due to Out-of-Memory (OOM) errors. Compressing a single modality offers only partial relief: visual compression alone extends the limit to 50K tokens, while textual compression

**Table 5:** Ablation study on training stages. We report both average and 128k scores on four tasks to show each stage's contribution. The full pipeline is marked with gray. **Bold** indicates the best.

| Training Stage | | | | VRAG | | NIAH | | ICL | | LongVLBench | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| S1 | S2 | S3 | S4 | avg | 128k | avg | 128k | avg | 128k | avg | >128k |
| ✓ | ✓ | ✓ | ✗ | 22.7 | 22.8 | **30.9** | **28.1** | 71.8 | 46.8 | 59.4 | 7.05 |
| ✓ | ✓ | ✗ | ✓ | **26.4** | 22.8 | 25.8 | 22.9 | 24.7 | 4.3 | 59.1 | 6.25 |
| ✗ | ✗ | ✓ | ✓ | 23.6 | 22.2 | 29.4 | 24.7 | 73.1 | 47.8 | 58.6 | **7.60** |
| ✓ | ✓ | ✓ | ✓ | 26.0 | **23.3** | 30.5 | 25.3 | **76.6** | **58.8** | **61.9** | **7.60** |

reaches 40K tokens, indicating that both modalities contribute significantly to memory pressure in ultra-long multimodal sequences. In stark contrast, our unified approach expands this limit to 120K tokens—**6-fold** the baseline capacity. This demonstrates that compressing both modalities in unison is not merely beneficial but essential for enabling direct training on ultra-long contexts.

**Table 4:** Impact of compression modules on maximum trainable sequence length before OOM (K = 1024 tokens).

| Compression | | Max. Len. |
|---|---|---|
| **Visual** | **Textual** | **(K tokens)** |
| ✗ | ✗ | 20 |
| ✓ | ✗ | 50 |
| ✗ | ✓ | 40 |
| ✓ | ✓ | **120** |

**Training Curriculum Analysis.** Tab. 5 reveals the distinct role of each stage. Comparing Rows 1 and 4, adding Stage 4 (long-context consolidation) substantially improves ICL-128k and LongVLBench average, confirming its importance for ultra-long reasoning. Row 2 shows that skipping Stage 3 (joint interleaved fine-tuning) causes ICL to collapse, indicating joint fine-tuning is essential for harmonizing both modalities. Row 3 demonstrates that even without compressor pretraining (Stages 1&2), the model achieves competitive LongVLBench scores, but the full pipeline (Row 4) yields the best overall balance across all tasks.

**Ablation on Chunk Size and Tokens per Chunk.** To reduce the hyperparameter search space, we tie the image and text parameters throughout this ablation, setting $C_v{=}C_t{=}C$ and $M_v{=}M_t{=}M$, leaving asymmetric configurations as future work. We ablate $C \in \{25, 50, 100\}$ and $M \in \{2, 4, 8\}$ under the full-layer injection strategy. Results are shown in Tab. 6. Fixing $M{=}4$ and varying $C$, we observe that smaller chunk sizes ($C = 25$) tend to improve performance on retrieval-centric tasks (VRAG and NIAH), as finer-grained compression preserves more local retrieval cues within each chunk. However, this advantage reverses at ultra-long contexts: $C{=}100$ achieves the best LongVLBench >128k score (7.60) and by far the highest ICL-128k score (58.8). We attribute this to the fact that larger chunks capture broader contextual patterns within each compressed unit, which is critical for tasks requiring global sequence understanding at extreme lengths. Fixing $C{=}100$ and varying $M$, we find $M{=}4$ to be the

**Table 6:** Ablation on chunk size and tokens per chunk. We set $C_v = C_t = C$ and $M_v = M_t = M$ across all configurations, and fix the full-layer injection strategy. The default configuration ($C$=100, $M$=4) is marked with gray. **Bold** indicates the best.

| $C$ | $M$ | VRAG | | NIAH | | ICL | | LongVLBench | |
|---|---|---|---|---|---|---|---|---|---|
| | | avg | 128k | avg | 128k | avg | 128k | avg | >128k |
| *Varying tokens per chunk $M$ (fixed $C$=100)* | | | | | | | | | |
| 100 | 2 | 28.7 | **29.0** | 27.8 | 25.0 | 69.1 | 39.3 | 58.8 | 7.45 |
| 100 | 4 | 26.0 | 23.3 | 30.5 | **25.3** | **76.6** | **58.8** | 61.9 | **7.60** |
| 100 | 8 | 26.7 | 23.1 | 31.1 | 25.0 | 64.1 | 20.5 | 60.3 | 7.15 |
| *Varying chunk size $C$ (fixed $M$=4)* | | | | | | | | | |
| 25 | 4 | **30.7** | 28.0 | 31.4 | 24.5 | 58.8 | 12.8 | **63.3** | 7.05 |
| 50 | 4 | 26.6 | 25.1 | **32.8** | 24.0 | 59.7 | 22.8 | 59.4 | 6.05 |
| 100 | 4 | 26.0 | 23.3 | 30.5 | **25.3** | **76.6** | **58.8** | 61.9 | **7.60** |
| *Joint variation* | | | | | | | | | |
| 50 | 8 | 29.7 | 27.2 | 31.6 | 23.4 | 49.0 | 7.5 | 62.1 | 7.00 |

**Table 7:** Injection layer strategy ablation for $C$=100, $M$=4. **Bold** denotes the best.

| Strategy | VRAG | | NIAH | | ICL | | LongVLBench | |
|---|---|---|---|---|---|---|---|---|
| | avg | 128k | avg | 128k | avg | 128k | avg | >128k |
| First-layer | **31.2** | **31.3** | **31.3** | **27.8** | 75.8 | 55.5 | 59.6 | 8.00 |
| First-half | 30.0 | 28.0 | 28.9 | 27.1 | 70.2 | 42.3 | 60.6 | 7.65 |
| Interval-2 | 28.9 | 27.8 | 30.6 | 27.2 | 70.8 | 39.3 | 61.8 | **8.25** |
| Full-layer | 26.0 | 23.3 | 30.5 | 25.3 | **76.6** | **58.8** | **61.9** | 7.60 |

optimal choice across all ultra-long metrics. Increasing to $M$=8 does not further improve performance and degrades ICL-128k significantly (20.5 vs. 58.8), while decreasing to $M$=2 reduces NIAH average (27.8 vs. 30.5) and ICL-128k (39.3 vs. 58.8), suggesting insufficient capacity to represent compressed chunk information. We also examine a joint variation ($C$=50, $M$=8), which maintains a higher compression rate than our default, but yields noticeably worse ICL performance (ICL avg: 49.0, ICL-128k: 7.5), further confirming that $C$ and $M$ have independent effects that cannot be captured by compression rate alone. We therefore adopt $C$=100, $M$=4 as our default configuration.

**Injection Layer Strategy.** Tab. 7 compares four injection strategies under our default $C$=100, $M$=4 setting. First-layer injection yields the strongest retrieval performance (VRAG avg: 31.2, NIAH avg: 31.3), as concentrating compressed features at the earliest layer provides rich context at the input stage. However, full-layer injection achieves the best ICL scores (ICL avg: 76.6, ICL-128k: 58.8) and LongVLBench average (61.9). While other strategies show competitive point scores in the longest bin, the superior average and ICL performance of full-layer

**Table 8:** Ablation on compressor architecture (fixed C=100, M=4, full-layer injection). The Q-Former compressor is marked with gray. **Bold** indicates the best.

| Compressor | VRAG | | NIAH | | ICL | | LongVLBench | |
|---|---|---|---|---|---|---|---|---|
| | avg | 128k | avg | 128k | avg | 128k | avg | >128k |
| Avg Pooling | **32.4** | **32.1** | 26.3 | 25.0 | 35.8 | 6.0 | 53.8 | 6.85 |
| Q-Former | 26.0 | 23.3 | **30.5** | **25.3** | **76.6** | **58.8** | **61.9** | **7.60** |

injection confirms its robustness. We attribute this to the holistic nature of these tasks: ICL requires inferring patterns from interleaved multimodal examples distributed across the entire context, while LongVLBench emphasizes comprehensive reasoning over long sequences. Both demand global context understanding that benefits from compressed representations being continuously available at every decoder layer, rather than concentrated at early layers alone.

**Compressor Architecture.** Tab. 8 compares our Q-Former compressor with a simpler average pooling baseline. Average pooling achieves stronger VRAG retrieval scores, as averaging preserves distributional cues beneficial for matching-based tasks. However, it severely underperforms on ICL and LongVLBench, indicating that average pooling lacks the capacity to selectively preserve structured information essential for reasoning over long contexts.

## 6 Conclusion

We introduced VLZip, a unified compression framework that enables efficient, high-fidelity reasoning over ultra-long multimodal sequences within a pure Transformer architecture. By hierarchically distilling visual and textual information and injecting it across all decoder layers, VLZip enables training on sequences up to 120K tokens—a 6× improvement—and processes contexts exceeding 280K tokens during inference. Its memory efficiency further enables scaling to 2M-token contexts, a scale infeasible for most comparable methods.

## Acknowledgements

This work is supported by Ant Group Research Fund and the National Natural Science Foundation of China (Grant No. 62472098), the Science and Technology Commission of Shanghai Municipality (No. 25511106100).

## References

1. Alayrac, J., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., Ring, R., Rutherford, E., Cabi, S., Han, T., Gong, Z., Samangooei, S., Monteiro, M., Menick, J.L., Borgeaud, S., Brock, A.,

Nematzadeh, A., Sharifzadeh, S., Binkowski, M., Barreira, R., Vinyals, O., Zisserman, A., Simonyan, K.: Flamingo: a visual language model for few-shot learning. In: Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022 (2022)
2. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., et al.: Qwen2.5-vl technical report. arXiv preprint arXiv:2502.13923 (2025)
3. Chen, L., Wei, X., Li, J., Dong, X., Zhang, P., Zang, Y., Chen, Z., Duan, H., Bin, L., Tang, Z., Yuan, L., Qiao, Y., Lin, D., Zhao, F., Wang, J.: Sharegpt4video: Improving video understanding and generation with better captions. In: Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024 (2024)
4. Chen, Z., Wang, W., Cao, Y., Liu, Y., Gao, Z., Cui, E., Zhu, J., Ye, S., Tian, H., Liu, Z., Gu, L., Wang, X., Li, Q., Ren, Y., Chen, Z., Luo, J., Wang, J., Jiang, T., Wang, B., He, C., Shi, B., Zhang, X., Lv, H., Wang, Y., Shao, W., Chu, P., Tu, Z., He, T., Wu, Z., Deng, H., Ge, J., Chen, K., Dou, M., Lu, L., Zhu, X., Lu, T., Lin, D., Qiao, Y., Dai, J., Wang, W.: Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. CoRR **abs/2412.05271** (2024). https://doi.org/10.48550/ARXIV.2412.05271
5. Cheng, X., Wang, X., Zhang, X., Ge, T., Chen, S.Q., Wei, F., Zhang, H., Zhao, D.: xrag: Extreme context compression for retrieval-augmented generation with one token. Advances in Neural Information Processing Systems **37**, 109487–109516 (2024)
6. Chevalier, A., Wettig, A., Ajith, A., Chen, D.: Adapting language models to compress contexts. In: The 2023 Conference on Empirical Methods in Natural Language Processing (2023)
7. Ge, T., Jing, H., Wang, L., Wang, X., Chen, S.Q., Wei, F.: In-context autoencoder for context compression in a large language model. In: The Twelfth International Conference on Learning Representations (2024)
8. 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)
9. Jiang, D., He, X., Zeng, H., Wei, C., Ku, M., Liu, Q., Chen, W.: Mantis: Interleaved multi-image instruction tuning. Trans. Mach. Learn. Res. **2024** (2024)
10. 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)
11. Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., Li, C.: Llava-onevision: Easy visual task transfer. Trans. Mach. Learn. Res. **2025** (2025), https://openreview.net/forum?id=zKv8qULV6n, accessed: 2026-06-25
12. Li, B., Ge, Y., Ge, Y., Wang, G., Wang, R., Zhang, R., Shan, Y.: Seed-bench: Benchmarking multimodal large language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13299–13308 (2024)
13. Li, J., Li, D., Savarese, S., Hoi, S.C.H.: BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In: International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii,

USA. Proceedings of Machine Learning Research, vol. 202, pp. 19730–19742. PMLR (2023)
14. Li, Y., Du, Y., Zhou, K., Wang, J., Zhao, W.X., Wen, J.: Evaluating object hallucination in large vision-language models. In: Bouamor, H., Pino, J., Bali, K. (eds.) Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023. pp. 292–305. Association for Computational Linguistics (2023). https://doi.org/10.18653/V1/2023.EMNLP-MAIN.20
15. Liao, Z., Wang, J., Yu, H., Wei, L., Li, J., Zhang, W.: E2llm: Encoder elongated large language models for long-context understanding and reasoning. In: Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing. pp. 19212–19241 (2025)
16. Liu, H., Li, C., Li, Y., Lee, Y.J.: Improved baselines with visual instruction tuning. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024. pp. 26286–26296. IEEE (2024)
17. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. In: Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023 (2023)
18. 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)
19. 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. In: The 36th Conference on Neural Information Processing Systems (NeurIPS) (2022)
20. Lu, S., Li, Y., Chen, Q.G., Xu, Z., Luo, W., Zhang, K., Ye, H.J.: Ovis: Structural embedding alignment for multimodal large language model. arXiv:2405.20797 (2024)
21. 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)
22. Meng, L., Yang, J., Tian, R., Dai, X., Wu, Z., Gao, J., Jiang, Y.G.: Deepstack: Deeply stacking visual tokens is surprisingly simple and effective for lmms. Advances in Neural Information Processing Systems **37**, 23464–23487 (2024)
23. Mu, J., Li, X., Goodman, N.: Learning to compress prompts with gist tokens. Advances in Neural Information Processing Systems **36**, 19327–19352 (2023)
24. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event. Proceedings of Machine Learning Research, vol. 139, pp. 8748–8763. PMLR (2021)
25. Shen, Y., Fu, C., Dong, S., Wang, X., Zhang, Y.F., Chen, P., Zhang, M., Cao, H., Li, K., Zheng, X., et al.: Long-vita: Scaling large multi-modal models to 1 million tokens with leading short-context accuracy. arXiv preprint arXiv:2502.05177 (2025)
26. Song, D., Chen, S., Chen, G.H., Yu, F., Wan, X., Wang, B.: Milebench: Benchmarking mllms in long context. CoRR **abs/2404.18532** (2024)
27. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information

Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA. pp. 5998–6008 (2017)
28. Wang, W., Zhang, S., Ren, Y., Duan, Y., Li, T., Liu, S., Hu, M., Chen, Z., Zhang, K., Lu, L., Zhu, X., Luo, P., Qiao, Y., Dai, J., Shao, W., Wang, W.: Needle in A multimodal haystack. In: Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024 (2024)
29. Wang, X., Song, D., Chen, S., Zhang, C., Wang, B.: Longllava: Scaling multi-modal llms to 1000 images efficiently via hybrid architecture. CoRR **abs/2409.02889** (2024)
30. Wang, Z., Yu, W., Ren, X., Zhang, J., Zhao, Y., Saxena, R., Cheng, L., Wong, G., See, S., Minervini, P., Song, Y., Steedman, M.: Mmlongbench: Benchmarking long-context vision-language models effectively and thoroughly. In: The 39th (2025) Annual Conference on Neural Information Processing Systems (2025), https://arxiv.org/abs/2505.10610, accessed: 2026-06-25
31. Xu, P., Ping, W., Wu, X., Xu, C., Liu, Z., Shoeybi, M., Catanzaro, B.: Chatqa 2: Bridging the gap to proprietary llms in long context and RAG capabilities. In: The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025. OpenReview.net (2025), https://openreview.net/forum?id=cPD2hU35x3, accessed: 2026-06-25
32. 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: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025. pp. 19792–19802. Computer Vision Foundation / IEEE (2025)
33. Ye, Z., Xia, K., Fu, Y., Dong, X., Hong, J., Yuan, X., Diao, S., Kautz, J., Molchanov, P., Lin, Y.C.: Longmamba: Enhancing mamba's long-context capabilities via training-free receptive field enlargement. In: The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025. OpenReview.net (2025)
34. 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)
35. Zeng, Q., Li, Y., Wang, Q., Jiang, P., Wu, Z., Cheng, M., Hou, Q.: A glimpse to compress: Dynamic visual token pruning for large vision-language models. CoRR **abs/2508.01548** (2025)
36. Zhang, K., Li, B., Zhang, P., Pu, F., Cahyono, J.A., Hu, K., Liu, S., Zhang, Y., Yang, J., Li, C., Liu, Z.: Lmms-eval: Reality check on the evaluation of large multimodal models. In: Findings of the Association for Computational Linguistics: NAACL 2025. p. 881–916. Association for Computational Linguistics (2025). https://doi.org/10.18653/v1/2025.findings-naacl.51
37. Zhou, K., Tang, Z., Ming, L., Zhou, G., Chen, Q., Qiao, D., Yang, Z., Qin, L., Qiu, M., Li, J., Zhang, M.: Mmlongcite: A benchmark for evaluating fidelity of long-context vision-language models. CoRR **abs/2510.13276** (2025)