VIFLOW-KV: VISUAL-FLOW GUIDED KV EVICTION FOR MULTIMODAL REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-horizon multimodal reasoning combines dense visual inputs with growing textual trajectories, making KV cache memory a major inference bottleneck for Large Multimodal Models (LMMs). KV cache eviction mitigates this burden by selectively retaining high-utility states under a limited cache budget. However, existing methods typically apply a modality-agnostic eviction policy, overlooking both the distinct retention requirements of visual and textual states and their evolving interaction during reasoning. Our attention analysis characterizes a heterogeneous *visual flow*: LMMs gradually shift the access to visual information from the original visual inputs toward later generated hidden states. Furthermore, later queries revisit both the original visual evidence and earlier generated states that carry visual information, revealing a long-range reuse of visual information. Guided by these observations, we propose **ViFlow-KV**, a visual-flow-guided KV eviction framework for multimodal reasoning. At each eviction step, ViFlow-KV assigns modality-specific budgets to visual and textual states, while incorporating visual-anchor signals to preserve reasoning-relevant visual evidence across evolving carriers. By preserving visual evidence across evolving visual and textual carriers, ViFlow-KV maintains critical context throughout long reasoning while keeping cache highly compressed. Extensive experiments show that ViFlow-KV consistently outperforms existing eviction methods, matching full attention performance at 2k token budget while achieving up to **3.05**$\times$ throughput speedup.

1 INTRODUCTION

Recent Large Multimodal Models (LMMs) have advanced from visual perception to multi-step reasoning over complex visual problems (Bai et al., 2025; Wang et al., 2025c; Yang et al., 2025b). This reasoning paradigm combines dense visual states with a continuously growing textual trajectory, substantially increasing the KV cache footprint during inference. For instance, with a batch size of 8, Qwen3-VL requires approximately 22.5 GiB for the KV cache alone when each sample contains 4K visual tokens and generates a 16K-token reasoning trajectory. This memory cost limits batch size and complicates the scalable deployment of long-horizon multimodal reasoning.

To alleviate this bottleneck, KV cache eviction has emerged as an effective approach by selectively retaining important KV states while reducing memory consumption. Most existing eviction methods are originally developed for text-only LLMs, exploiting the observation that cached states contribute unequally to future generation. They reduce redundancy in long input contexts during prefill (Li et al., 2024; Cai et al., 2024; Feng et al., 2025; Xiao et al., 2025) or in growing reasoning trajectories during decoding (Cai et al., 2025; Mao et al., 2026; Guo et al., 2026). In parallel, fine-grained visual representations introduce an additional source of redundancy in LMMs. Visual representations exhibit substantial spatial redundancy (Chen et al., 2024a; Yang et al., 2025a; Xing et al., 2024; Wang et al., 2025b), with task-relevant information typically localized to a query-dependent subset of visual tokens (Khaki et al., 2025; Su et al., 2026; Hong et al., 2026).

Despite this progress, existing approaches typically address either input-side visual redundancy or text-oriented KV eviction, without jointly modeling the retention of visual and textual states throughout reasoning. However, visual and textual KV states exhibit markedly different importance distributions, indicating that a retention criterion calibrated for one modality may not transfer reliably to the other. Beyond this numerical discrepancy, these two modalities play distinct roles during

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

reasoning: visual states are revisited intermittently, while textual states accumulate throughout the reasoning trajectory. Incorrect eviction would irreversibly discard task-relevant evidence and increase the risk of downstream reasoning errors. These observations motivate a key question: ***how do visual and textual states interact throughout multimodal reasoning***, and ***what implications does this interaction have for KV cache eviction***?

To better understand this interaction, we conduct a fine-grained analysis of cross-modal attention dynamics throughout decoding. As shown in Figure 1, attention to the original visual states generally decreases during reasoning as textual states accumulate and exhibit stronger intra-modal similarity. However, this aggregate trend masks the continued role of visual evidence. As indicated in Figure 2, salient visual activation accompanies the generation of textual states expressing visually grounded facts. Moreover, when the evidence becomes relevant again, later reasoning tends to activate these related textual states. Together, these patterns reveal an evolving ***visual flow***: visual evidence acquires textual carriers during reasoning, while later steps jointly revisit the original evidence and its related carriers, allowing the evidence to propagate through reasoning. Overall, effective KV eviction should preserve reasoning-relevant evidence across these evolving carriers rather than treating visual/textual retention independently.

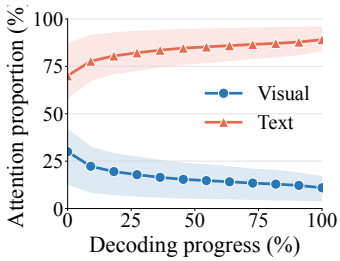


Figure 1: Attention shifts from visual tokens to historical text as reasoning progresses. Results are based on 100 full trajectories generated by Qwen3-VL.

Building on these insights, we propose **ViFlow-KV**, a visual-flow-guided KV cache eviction method that jointly adapts budget allocation and importance estimation throughout multimodal reasoning. For budget allocation, we introduce Evolving Modality-Demand Allocation (EMDA), which uses attention mass from the query prototype as a lightweight proxy to estimate the current visual and textual demand of each KV head and dynamically adjusts their modality-specific budgets at every eviction step. This allows the cache allocation to follow both temporal shifts in modality demand and head-specific modality preferences simultaneously. For importance estimation, we introduce visual-anchor importance estimation (VAIE) to support accurate and visually grounded reasoning. Specifically, VAIE enriches importance estimation with queries that strongly attend to visual context, providing a visually informed perspective to preserve perceptual evidence across original visual states and related textual representations. Extensive experiments across six multimodal reasoning benchmarks show that ViFlow-KV consistently outperforms existing eviction methods. Particularly, ViFlow-KV achieves performance comparable to the full-attention baseline at a 2K-token budget and achieves up to **3.05**$\times$ throughput speedup under long-horizon generation.

Our contributions are summarized as follows: **1)** We characterize visual flow in multimodal reasoning, revealing how visual evidence extends into textual states and is later revisited across both modalities. **2)** We propose ViFlow-KV, a training-free framework that combines evolving modality-demand allocation with visual-anchor importance estimation to enable efficient but accurate reasoning under constrained cache budgets. **3)** Extensive experiments across six benchmarks show that ViFlow-KV consistently outperforms representative baselines under constrained cache budgets, substantially reducing cache usage with minimal accuracy loss.

2 Motivation Analysis

2.1 Preliminary

During multimodal reasoning, given a multimodal input $x = \{p, I\}$, where p denotes the prompt and I the visual input, an LMM generates a reasoning trajectory and a final answer. At decoding step t, the attention score from query $\mathbf{q}_t^{l,h}$ to a visible key $\mathbf{k}_j^{l,h}$ is computed as:

$$A_{t,j}^{l,h} = \frac{\exp\left(\langle \mathbf{q}_t^{l,h}, \mathbf{k}_j^{l,h} \rangle / \sqrt{d}\right)}{\sum_{u \in \mathcal{V} \cup \mathcal{T}_t} \exp\left(\langle \mathbf{q}_t^{l,h}, \mathbf{k}_u^{l,h} \rangle / \sqrt{d}\right)}, \tag{1}$$

where l and h denote the layer and attention head, respectively. The sets $\mathcal{V}$ and $\mathcal{T}_t$ index the visual and textual states in the cached context, respectively. At the modality level, we quantify the attention

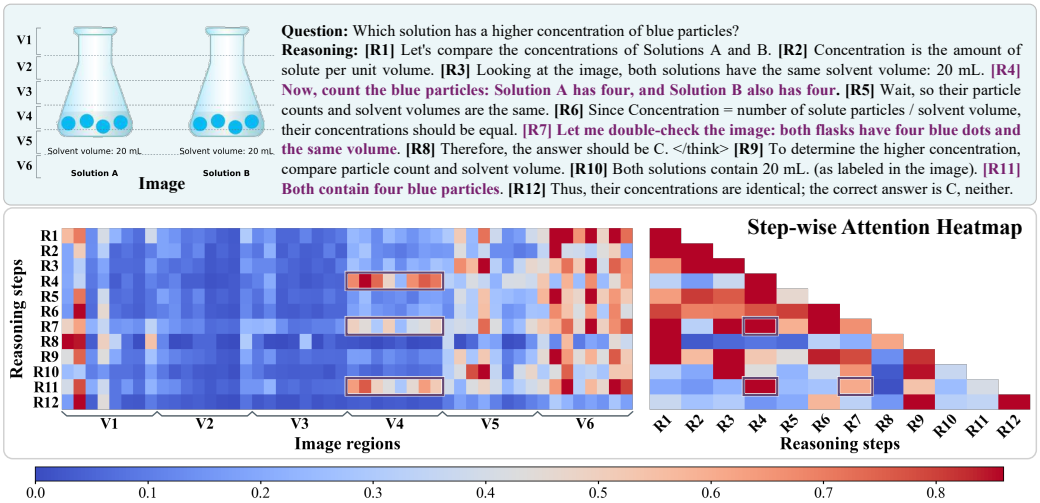


Figure 2: **Visual flow during multimodal reasoning.** In this representative trajectory, R4 identifies the blue-particle count in region V4 (the blue particles in both flasks). When R7 double-checks the particle count and solvent volume, it attends to both V4 and R4. Later, R11 confirms that both solutions contain four blue particles, attending to V4 together with R4 and R7. This pattern suggests that earlier reasoning steps can serve as textual carriers of visual evidence that later steps revisit.

allocation to visual and textual states through their respective attention masses: $M^{l,h}_{t,\mathcal{V}} = \sum_{j\in\mathcal{V}} A^{l,h}_{t,j}$ and $M^{l,h}_{t,\mathcal{T}} = \sum_{j\in\mathcal{T}_t} A^{l,h}_{t,j} = 1 - M^{l,h}_{t,\mathcal{V}}$. As shown in Figure 1, visual attention mass gradually declines as the textual context expands during reasoning. However, this aggregate trend does not fully characterize the interaction between visual and textual states or reveal whether visual states can be effectively compressed. To better understand this interaction, we further trace how visual evidence is carried and revisited across visual and textual states throughout reasoning.

2.2 Visual Flow In Multimodal Reasoning

Visual evidence can contribute to textual state formation during reasoning. Consider a typical Transformer block and let $\mathcal{V}_e$ denote the visual positions associated with evidence e. At a textual position g where the $\mathcal{V}_e$ is first activated, the explicit visual contribution can be written as:

$$\mathbf{h}^{l+1}_g = W^l_O \sum_{j\in\mathcal{V}_e} A^l_{g,j}\mathbf{v}^l_j + \mathbf{r}^l_g, \tag{2}$$

where W^l_O is the attention output projection and $\mathbf{r}^l_g$ collects other terms from positions outside $\mathcal{V}_e$. This contribution is then reflected in the next-layer key and value representations. Consider the key representation, which participates in subsequent query matching:

$$\mathbf{k}^{l+1}_g = RoPE(W_K\, h^{l+1}_g, g) = RoPE(W_K \left[W^l_O \sum_{j\in\mathcal{V}_e} A^l_{g,j}\mathbf{v}^l_j + \mathbf{r}^l_g \right], g), \tag{3}$$

where W_K denotes the key projection. Eq. 3 identifies a computational pathway that visual evidence can contribute to textual states and shapes their subsequent representations. In other words, generated states can serve as additional carriers of visual evidence.

Visual evidence propagates through successive carriers. Textual carriers provide additional states through which subsequent reasoning can access visually grounded information from these carriers. Let $\mathcal{T}_{e,t}$ denote the historical textual positions expressing evidence e. At a later reasoning position t where the $\mathcal{V}_e$ is reactivated, the new state can be decomposed as:

$$\mathbf{h}^{l+1}_t = W^l_O \left(\sum_{j\in\mathcal{V}_e} A^l_{t,j}\mathbf{v}^l_j + \sum_{g\in\mathcal{T}_{e,t}} A^l_{t,g}\mathbf{v}^l_g \right) + \mathbf{r}^l_t, \tag{4}$$

where $\mathbf{r}_t^l$ collects the remaining terms. Compared with Eq. 2, Eq. 4 introduces an additional text-mediated pathway where later reasoning jointly accesses original visual states and visually grounded textual states, allowing visual evidence to propagate through successive carriers.

To quantify these pathways in actual reasoning, we perform fine-grained annotation on 100 examples by pairing visual regions with their corresponding textual spans along each reasoning trajectory. Because each reasoning trajectory involves multiple visual regions, for each visual region, textual spans are categorized as those expressing the same visual fact (*Matched*), other visual facts (*Unmatched*), or no visual facts (*Non-visual*). We then compare key similarity between visual evidence and its associated textual carriers to examine the representational relationship in Eq. (3), and analyze textual attention conditioned on different activation thresholds to quantify the revisitation pattern in Eq. (4). More details are provided in Appendix B.1.

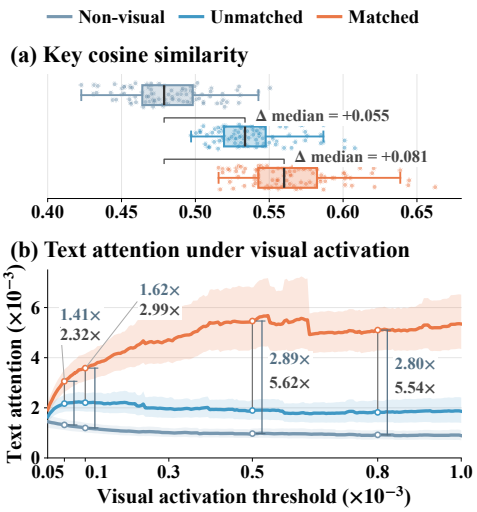


Figure 3: Visual states forms textual carriers and propagates through joint reasoning. (a) Matched textual states show higher key similarity to the corresponding visual states than other states. (b) Stronger visual activation is accompanied by substantially higher attention to matched textual states.

Figure 3 shows quantitative results supporting both stages of visual flow. In Figure 3(a), matched textual states show the strongest alignment with visual evidence, reaching a median key similarity of 0.56, which is 0.081 higher than non-visual states. In Figure 3(b), matched states consistently receive higher attention across visual activation thresholds, with the gap exceeding **5.5**× over non-visual states at the highest threshold. Furthermore, when visual evidence is activated (e.g., at the activation threshold of 5×10^{-4}), its matched textual carriers receive substantially stronger activation (i.e., 5×10^{-3}), exceeding **10**× the attention response of original visual states at higher thresholds. As reasoning progresses, the accumulated textual carriers would become increasingly dominant pathways for visual evidence access, as the expanding textual trajectory gradually shapes the model's attention allocation. Figure 2 further illustrates this recurrent pattern: visual evidence first accessed at R4 is revisited with its textual carrier at R7, and later with both earlier carriers at R11. Together, these observations characterize an evolving *visual flow*: visual evidence is progressively reflected in textual representations and later reasoning reconnects both original evidence and its accumulated carriers. Therefore, effective KV eviction could progressively reduce visual token redundancy while preserving critical perceptual evidence across evolving carriers to support accurate reasoning.

3 ViFlow-KV

3.1 Overview

Motivated by the visual flow identified above, we propose **ViFlow-KV**, a training-free KV cache eviction framework for multimodal reasoning. As illustrated in Figure 4, ViFlow-KV comprises two complementary components: Evolving Modality Demand Allocation (EMDA) and Visual-Anchor Importance Estimation (VAIE). At each eviction, EMDA adjusts visual and textual allocations according to head-specific attention demand, capturing both temporal demand shifts and inherent modality preferences within the model. Complementarily, VAIE selects visual-intensive reasoning states as visual anchors and incorporates their queries to estimate historical KV importance. Together, EMDA determines how much capacity each modality receives, while VAIE determines which KV entries to retain. We describe the details of each module below.

3.2 KV Eviction Formulation

Following window-based KV eviction methods for reasoning (Cai et al., 2025; Mao et al., 2026), ViFlow-KV checks the cache every W decoding steps and performs eviction when it exceeds the retention budget B. For clarity, we omit the layer index in this section. For each head h, the r most

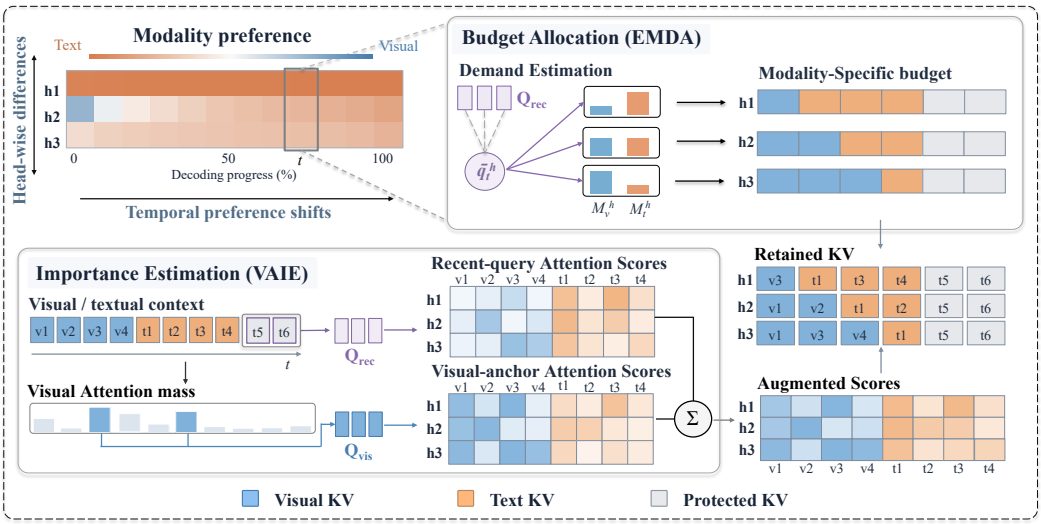


Figure 4: **Overview of ViFlow-KV.** EMDA dynamically allocates modality-specific KV budgets for each head according to evolving visual and textual demand, while VAIE combines recent-query and visual-anchor signals to estimate the importance of historical states. The resulting modality budgets and augmented importance scores are jointly used for budget-constrained KV selection.

recent positions are protected to preserve local reasoning context, denoted by $\mathcal{R}_t^h$. The remaining historical candidates are:

$$\mathcal{H}_t^h = \mathcal{C}_t^h \setminus \mathcal{R}_t^h = \mathcal{V}_t^h \cup \mathcal{T}_t^h, \tag{5}$$

where $\mathcal{C}_t^h$ denotes all cached positions, $\mathcal{V}_t^h$ and $\mathcal{T}_t^h$ denote visual and textual states, respectively. Under modality-agnostic allocation, all historical candidates share the retention budget $B - r$. Following R-KV (Cai et al., 2025), we aggregate attention from recent queries $\mathcal{Q}_{t,\text{rec}}^h = \{\mathbf{q}_i^h \mid i \in \mathcal{R}_t^h\}$ as $A_{t,j}^{h,\text{rec}}$. The retention score and selected cache are computed as:

$$\Phi_{t,j}^h = \lambda A_{t,j}^{h,\text{rec}} - (1-\lambda)\rho_{t,j}^h, \qquad \mathcal{S}_t^h = \mathcal{R}_t^h \cup \text{TopK}_{B-r}\left(\mathcal{H}_t^h; \Phi_{t,\cdot}^h\right), \tag{6}$$

where $\rho_{t,j}^h$ denotes key-similarity redundancy and λ balances importance and redundancy. ViFlow-KV adapts both components to visual flow: EMDA splits retention budget across modalities, while VAIE fuses importance signals from recent and visual-anchor queries into $A_{t,j}^{h,\text{aug}}$, yielding:

$$\Phi_{t,j}^h = \lambda A_{t,j}^{h,\text{aug}} - (1-\lambda)\rho_{t,j}^h, \qquad \mathcal{S}_t^h = \mathcal{R}_t^h \cup \text{TopK}_{B_{t,\mathcal{V}}^h}\left(\mathcal{V}_t^h; \Phi_{t,\cdot}^h\right) \cup \text{TopK}_{B_{t,\mathcal{T}}^h}\left(\mathcal{T}_t^h; \Phi_{t,\cdot}^h\right), \tag{7}$$

where $B_{t,\mathcal{V}}^h + B_{t,\mathcal{T}}^h = B - r$. The two components are detailed in the following subsections.

3.3 Evolving Modality-Demand Allocation

Existing KV allocation methods typically adopt either fixed budget allocation (Wang et al., 2025b; Cai et al., 2024) or modality-agnostic allocation (Feng et al., 2025). However, visual flow reveals that the relative demand for visual and textual states evolves throughout reasoning. Furthermore, we observe that modality demand also differs across heads (see Appendix A.2), consistent with prior observations of heterogeneous modality preferences among attention heads (Wu et al., 2025; Kang et al., 2025b). These temporal and head-specific variations motivate EMDA to dynamically allocate the historical budget across modalities for each KV head at every eviction step.

Motivated by the more concentrated distribution of pre-RoPE queries (Mao et al., 2026; Luo et al., 2026), EMDA estimates the current modality demand using a centroid query. Specifically, we compute the center of r most recent queries in pre-RoPE space and rotate it to the current position t:

$$\bar{\mathbf{q}}_{t,\text{pre}}^h = \frac{1}{r}\sum_{i \in \mathcal{R}_t^h} \mathbf{q}_{i,\text{pre}}^h, \qquad \hat{\mathbf{q}}_t^h = \text{RoPE}\left(\bar{\mathbf{q}}_{t,\text{pre}}^h, t\right). \tag{8}$$

Using this centroid query, we compute attention $\widehat{A}_{t,j}^h$ over the cached keys and aggregate it by modality as $\widehat{M}_{t,\mathcal{V}}^h = \sum_{j \in \mathcal{V}_t^h} \widehat{A}_{t,j}^h$ and $\widehat{M}_{t,\mathcal{T}}^h = \sum_{j \in \mathcal{T}_t^h} \widehat{A}_{t,j}^h$. These attention masses serve as lightweight

proxies for the relative modality demand. Accordingly, EMDA allocates the historical budget as:

$$B^h_{t,\mathcal{V}} = (B-r)\frac{\widehat{M}^h_{t,\mathcal{V}}}{\widehat{M}^h_{t,\mathcal{V}} + \widehat{M}^h_{t,\mathcal{T}}}, \qquad B^h_{t,\mathcal{T}} = B - r - B^h_{t,\mathcal{V}}. \tag{9}$$

This allocation dynamically adapts the historical budget to the relative visual and textual demand of each head at every eviction step. In this way, EMDA captures both head-specific modality preferences and their evolution throughout reasoning.

3.4 Visual-Anchor Importance Estimation

Given the modality-specific budgets, the remaining challenge is to identify which historical states should be retained. Visual-intensive textual states preserve visually grounded information and can serve as textual carriers of visual evidence, making them important candidates for retaining perceptual evidence. Motivated by these observations, we propose VAIE, which jointly leverages recent reasoning context and visual-anchor signals to estimate historical state importance and preserve critical evidence across evolving carriers.

At each eviction step, VAIE identifies visual-intensive reasoning states as visual anchors. To reduce the influence of individual head preferences (Wan et al., 2025; Kang et al., 2025a), we average the visual attention mass in each layer and select the top-m positions from current window W with the strongest visual activation:

$$\bar{M}_{\tau,\mathcal{V}} = \frac{1}{H}\sum_{h=1}^{H} M^h_{\tau,\mathcal{V}}, \qquad \mathcal{A}_{t,\mathrm{vis}} = \mathrm{TopK}_m\left(\bar{M}_{\tau,\mathcal{V}}\right). \tag{10}$$

where H denotes the number of attention heads. The corresponding queries form the visual-anchor query set $\mathcal{Q}^h_{t,\mathrm{vis}} = \{\mathbf{q}^h_\tau \mid \tau \in \mathcal{A}_{t,\mathrm{vis}}\}$. Using the recent query set in Eq. 6 and the visual-anchor query set, we compute the attention scores $A^{h,\mathrm{rec}}_{t,j}$ and $A^{h,\mathrm{vis}}_{t,j}$, and equally fuse them as

$$A^{h,\mathrm{aug}}_{t,j} = \frac{A^{h,\mathrm{rec}}_{t,j} + A^{h,\mathrm{vis}}_{t,j}}{2}. \tag{11}$$

The fused score $A^{h,\mathrm{aug}}_{t,j}$ is used in Eq. 7, while the redundancy term remains unchanged. This fusion allows token importance to reflect both the current reasoning context and visually grounded evidence that remain useful for later reasoning.

4 Experiments

4.1 Experimental Setup

To evaluate ViFlow-KV on general multimodal reasoning, we conduct experiments on six widely adopted benchmarks (Lu et al., 2024; Yu et al., 2024; Chen et al., 2024b; Yue et al., 2024; Li et al., 2026; Wang et al., 2024) and compare against SnapKV (Li et al., 2024), R-KV (Cai et al., 2025), and Ada-KV (Feng et al., 2025). SnapKV and R-KV are representative importance-based eviction methods that use a uniform cache budget across attention heads. To evaluate adaptive budget allocation, we further apply Ada-KV at each eviction step to redistribute the cache budget across heads according to the current context. However, as a modality-agnostic method, Ada-KV does not explicitly model the evolving demand for visual and textual states during multimodal reasoning. For a controlled comparison, all methods use the same cache budget and eviction schedule. We set the number of protected recent positions to $r = 8$, the number of visual anchors to $m = 8$, and the generation window length to $W = 128$. The trade-off coefficient λ in the Eq. 7 is set to $\lambda = 0.1$.

4.2 Main Results

We evaluate all methods under four constrained cache budgets spanning different compression levels. Each image is standardized to 2,048 visual tokens, matching the largest budget and ensuring that any subsequent text generation triggers eviction. Under this setting, even the largest budget of 2,048 provides only enough cache capacity to accommodate the visual input itself, leaving no additional

Table 1: **Main results under aggressive cache budgets (Qwen3-VL-4B Thinking).** We compare ViFlow-KV with prior representative methods across six benchmarks. MVista, CXiv RQ, and MMrL denote MathVista (Lu et al., 2024), CharXiv RQ (Wang et al., 2024), and MMr-Life (Li et al., 2026), respectively. B denotes the per-head KV cache budget. Avg. is the average performance across six benchmarks and Rec. denotes performance recovery relative to FullKV performance.

Method	MMStar	MVista	CXiv RQ	MMVet	MMMU	MMrL	Avg.	Rec.
FullKV	70.80	78.80	53.80	65.78	65.22	32.20	61.10	100.00
$B = 256$								
SnapKV	52.40	48.30	23.20	51.93	31.11	9.27	36.04	58.99
R-KV	52.60	51.60	26.90	50.23	29.33	8.78	36.57	59.85
Ada-RKV	52.30	51.10	24.70	50.60	28.22	13.17	36.68	60.03
ViFlow-KV	**55.47**	**55.40**	**28.80**	**54.68**	**32.44**	**21.95**	**41.46**	**67.86**
$B = 512$								
SnapKV	59.53	60.50	34.70	59.27	39.67	17.07	45.12	73.85
R-KV	59.20	62.20	34.80	57.43	40.89	20.98	45.92	75.16
Ada-RKV	60.27	62.00	33.90	54.86	41.11	18.54	45.11	73.83
ViFlow-KV	**63.93**	**66.60**	**40.20**	**62.89**	**46.78**	**25.37**	**50.96**	**83.40**
$B = 1,024$								
SnapKV	65.13	68.20	44.00	63.72	47.33	26.83	52.54	85.99
R-KV	65.80	68.90	43.60	61.06	50.44	21.95	51.96	85.04
Ada-RKV	66.33	71.20	44.50	60.96	52.11	23.41	53.09	86.89
ViFlow-KV	**67.87**	**73.40**	**45.80**	**64.54**	**54.89**	**27.80**	**55.72**	**91.19**
$B = 2,048$								
SnapKV	69.87	74.10	51.40	63.89	56.44	26.83	57.09	93.44
R-KV	70.67	76.00	50.70	61.10	61.37	22.44	57.05	93.37
Ada-RKV	70.47	75.80	51.80	61.79	61.00	24.39	57.54	94.17
ViFlow-KV	**71.47**	**78.00**	**53.30**	**64.50**	**62.11**	**31.22**	**60.10**	**98.36**

space for subsequent reasoning states. This also makes $B = 2,048$ a particularly meaningful setting, as strong performance under this budget indicates that long-horizon reasoning can be sustained without expanding the KV cache beyond the memory scale of the original visual input.

As shown in Table 1, ViFlow-KV consistently outperforms existing eviction methods across all four cache budgets. Across the four settings, ViFlow-KV improves the six-benchmark average over the strongest baseline by 3.75 points on average. The gains are particularly pronounced under aggressive compression, reaching 4.78 and 5.04 points at budgets of 256 and 512, respectively. Notably, at $B = 2,048$, which matches the number of input visual tokens, ViFlow-KV recovers 98.36% of the average FullKV performance. Furthermore, on first four single-image benchmarks, where this budget matches the visual-input footprint, the recovery further reaches 99.29%, with an average gap of only 0.48 points from FullKV. MMMU and MMr-Life impose a stricter compression regime because their multi-image visual context. However, ViFlow-KV remains the strongest eviction method on both benchmarks. This result suggests that near-FullKV performance can be sustained while keeping KV cache within the visual-input footprint rather than allowing it to grow with the reasoning trajectory, offering a promising direction for memory-efficient long-horizon multimodal reasoning.

The consistent gains of ViFlow-KV further reveal the limitations of existing eviction strategies. R-KV and SnapKV focus on importance-based state selection under fixed per-head budgets, without explicitly modeling the evolving interaction between visual and textual states. Ada-RKV further redistributes the cache budget across heads within each layer to capture head-wise differences in cache demand, but this introduces unequal per-head cache lengths and requires additional support for variable-length attention. Despite this added flexibility, its gains over R-KV remain inconsistent. In contrast, ViFlow-KV does not rely on cross-head budget redistribution, yet consistently outperforms Ada-RKV across all four budgets. Overall, by preserving visual evidence across its evolving visual and textual carriers, ViFlow-KV achieves more consistent performance across different budgets.

4.3 Ablation Study

We conduct component ablations at $B = 512$, where the constrained cache budget makes eviction decisions more consequential. As shown in Table 2, EMDA and VAIE independently improve the

Table 2: **Ablation study of ViFlow-KV under a 512-token KV cache budget.** We analyze the effect of different components used for ViFlow-KV on MMStar, CharXiv RQ, and MathVista. Avg. denotes the unweighted average across the three benchmarks.

Variant	EMDA	VAIE	MMStar	CharXiv RQ	MathVista	Avg.
Baseline	✗	✗	59.20	34.80	62.20	52.07
+ EMDA	✓	✗	61.93	39.40	64.10	55.14
+ VAIE	✗	✓	60.33	39.90	64.60	54.94
ViFlow-KV	✓	✓	**63.93**	**40.20**	**66.60**	**56.91**

Table 3: **System efficiency of ViFlow-KV under long-horizon multimodal generation.** We report the maximum supported batch size and end-to-end throughput under two long-context workloads. Speedup is measured relative to FullKV under the same workload.

Input Length		Gen. Length	Method	Budget	Batch	Throughput (tok/s)	Speedup
Visual	Text						
2K	512	8K	FullKV	Full	42	1,087.1	1.00×
			ViFlow-KV	512	**436**	**4,309.9**	**3.96**×
			ViFlow-KV	2,048	**140**	**1,767.3**	**1.63**×
4K	512	16K	FullKV	Full	21	575.5	1.00×
			ViFlow-KV	512	**434**	**4,093.2**	**7.11**×
			ViFlow-KV	2,048	**141**	**1,757.1**	**3.05**×

baseline across three benchmarks by 3.07 and 2.87 points on average, respectively. Combining both components raises the average score from 52.07 to 56.91 and achieves the best performance on all three benchmarks. These results demonstrate that the two components of ViFlow-KV complement each other by modeling both evolving modality demand and visual evidence across evolving carriers.

(1) EMDA captures evolving modality demand. Visual and textual demand varies across reasoning steps and heads, so modality-agnostic allocation can be suboptimal. EMDA improves MMStar, CharXiv RQ, and MathVista by 2.73, 4.60, and 1.90 points, respectively, with an average gain of 3.07 points, validating the benefit of dynamically adapting modality-specific budgets.

(2) VAIE preserves visual evidence beyond recent context. Although recent queries effectively capture local reasoning context (Yu et al., 2025), they may miss visual evidence that becomes relevant again after several reasoning steps. Adding VAIE improves MMStar, CharXiv RQ, and MathVista by 1.13, 5.10, and 2.40 points, respectively, yielding an average gain of 2.87 points. This supports preserving visual evidence across its evolving carriers during long-horizon reasoning.

4.4 More Analysis

Efficiency Analysis We evaluate the system efficiency of ViFlow-KV under two long-horizon multimodal generation workloads, as shown in Table 3. At $B = 2,048$, ViFlow-KV achieves **1.63**× and **3.05**× end-to-end throughput speedups under the two workloads, respectively. Combined with its near-FullKV accuracy under the same cache budget, these results demonstrate that ViFlow-KV can improve inference efficiency without requiring aggressive cache compression. When the budget is further reduced to $B = 512$, the speedups increase to **3.96**× and **7.11**×, showing the additional efficiency enabled by tighter cache constraints. Overall, these results demonstrate the scalability and practical deployability of ViFlow-KV for long-horizon multimodal reasoning.

Generalization across different LMMs We evaluate LMMs from different families and scales to assess the generalization of ViFlow-KV. As shown in Table 4, ViFlow-KV consistently outperforms R-KV across both model scales and architectures. Under a 512-token budget, ViFlow-KV improves the average performance over R-KV by 3.39 points on Qwen3-VL-8B-Thinking and 2.55 points on InternVL-3.5-8B-Thinking. Notably, on InternVL-3.5-8B-Thinking, ViFlow-KV retains 95.87% of FullKV performance on average, demonstrating strong performance retention under substantial cache compression. On MMr-Life, ViFlow-KV achieves 24.88, even surpassing the FullKV result of 23.90. Overall, these results demonstrate that the visual-flow aware design transfers effectively across distinct LMM families and architectures without model-specific adaptation.

Table 4: **Generalization across model scales and architectures under a 512-token KV cache budget.** We evaluate ViFlow-KV across different model scales and architectures. R-KV (Cai et al., 2025) is included for comparison. Avg. denotes the average performance across four benchmarks, and Rec. denotes performance recovery relative to FullKV. The best eviction result is shown in bold.

Method	MMStar	CharXiv RQ	MathVista	MMr-Life	Avg.	Rec.
Qwen3-VL-8B-Thinking						
FullKV	74.93	52.60	79.20	32.68	59.85	100.00
R-KV	64.53	38.60	67.70	24.88	48.93	81.75
ViFlow-KV	**68.73**	**41.60**	**72.60**	**26.34**	**52.32**	**87.42**
InternVL-3.5-8B-Thinking						
FullKV	66.40	46.10	77.50	23.90	53.48	100.00
R-KV	65.73	40.30	65.90	22.93	48.72	91.10
ViFlow-KV	**66.00**	**43.30**	**70.90**	**24.88**	**51.27**	**95.87**

5 Related Work

Multimodal Reasoning LMMs (Bai et al., 2025; Wang et al., 2025c; Hong et al., 2025; Team et al., 2026) have advanced beyond visual perception toward multi-step reasoning over complex multimodal inputs. Recent studies have shown that effectively attending to and integrating visual evidence throughout inference is critical for multimodal reasoning (Jiarui Zhang et al., 2025; Huang et al., 2025; Wang et al., 2026; Liu et al., 2026; Yang et al., 2026). Specifically, VPPO (Huang et al., 2025) prioritizes visually dependent tokens during policy optimization, while VGPO (Wang et al., 2026) reinforces visual attention across different reasoning steps. Despite these advances, stronger reasoning often relies on longer generation trajectories, substantially increasing inference costs on challenging problems (Guo et al., 2025; Yang et al., 2025b; Dai et al., 2026; Wang et al., 2025a). Yet existing work remains centered on reasoning accuracy, leaving how input visual evidence and generated states interact and evolve throughout reasoning insufficiently explored. This gap is essential for efficient multimodal reasoning, where a limited cache must accommodate both dense visual evidence and an expanding textual trajectory.

KV Cache Eviction KV cache eviction typically comprises two core components: importance estimaton for historical states (Li et al., 2024; Ahn et al., 2026; Dong et al., 2026; Liu et al., 2025) and retention-budget allocation across the model (Feng et al., 2025; Du et al., 2025; Fu et al., 2025; Wan et al., 2024; Wang et al., 2025b). For importance estimation, SnapKV (Li et al., 2024) identifies important states using a recent observation window, whereas R-KV (Cai et al., 2025) further penalizes token redundancy among reasoning states. Recent reasoning-aware methods (Hu et al., 2025; Ramachandran et al., 2025) also exploit attention patterns specific to textual reasoning trajectories. For budget allocation, SparseMM (Wang et al., 2025b) identifies visually active attention heads and statically assigns them larger KV-cache budgets. Ada-KV (Feng et al., 2025), PyramidKV (Cai et al., 2024), and CAKE (Qin et al., 2025) explore non-uniform allocation across heads or layers. These methods estimate cache demand from prefill-stage information, without considering its evolution throughout reasoning. In this work, we address above limitations by modeling visual flow and adapting KV cache allocation and importance estimation accordingly.

6 Conclusion

In this work, we identify an evolving visual flow, where modality demand shifts throughout reasoning and visual evidence could be revisited through both visual states and textual carriers. Building on this finding, we introduce ViFlow-KV, a training-free KV cache eviction framework for efficient multimodal reasoning. Specifically, ViFlow-KV combines EMDA to adapt visual–textual cache allocation to evolving modality demand with VAIE to preserve relevant visual evidence across evolving carriers. Extensive experiments on six benchmarks show that ViFlow-KV consistently outperforms existing eviction methods under constrained budgets and generalizes across model scales and architectures. Notably, with a 2,048-token budget matching input visual tokens, ViFlow-KV recovers 98.36% of FullKV performance on average while achieving up to **3.05**× end-to-end throughput speedup. These results demonstrate that near-FullKV performance can be maintained with improved efficiency while preventing KV-cache growth with reasoning length, providing a promising direction for scalable and memory-efficient long-horizon multimodal reasoning.

AI USE STATEMENT

Generative AI tools were used primarily to assist with language polishing and improving the clarity of the manuscript. In addition, GPT-6-Astra assisted with visual–textual annotation, which was manually reviewed by human experts, as detailed in Appendix B.1. They were not used to generate or fabricate references, and all cited literature was independently verified by the authors. Moreover, all research ideas, method design, experimental implementation, data analysis, and scientific conclusions were developed and verified by the authors.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the ViFlow-KV formulation, implementation, hyperparameters, cache budgets, and evaluation settings in the main paper and Appendix. All experiments are conducted on publicly available models and benchmarks, with consistent preprocessing and evaluation protocols across compared methods. We also report the configurations used for efficiency evaluation and additional implementation details required to reproduce our results. To further facilitate reproducibility, we will release the implementation and evaluation scripts upon publication.

REFERENCES

Jinwoo Ahn, Ingyu Seong, Akhil Kedia, Junhan Kim, Hyemi Jang, Kangwook Lee, and Yongkweon Jeon. Lookaheadkv: Fast and accurate kv cache eviction by glimpsing into the future without generation. *arXiv preprint arXiv:2603.10899*, 2026.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Zefan Cai, Yichi Zhang, Bofei Gao, Yuliang Liu, Tianyu Liu, Keming Lu, Wayne Xiong, Yue Dong, Baobao Chang, Junjie Hu, and Wen Xiao. Pyramidkv: Dynamic kv cache compression based on pyramidal information funneling. *arXiv preprint arXiv:2406.02069*, 2024.

Zefan Cai, Wen Xiao, Hanshi Sun, Yikai Zhang, Ke Wan, Yucheng Li, Yeyang Zhou, Li-Wen Chang, Jiuxiang Gu, Zhen Dong, et al. R-kv: Redundancy-aware kv cache compression for reasoning models. *Advances in neural information processing systems*, 38:60980–61005, 2025.

Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. 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 (ECCV)*, pp. 19–35. Springer, 2024a.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating Large Vision-Language Models? *Advances in Neural Information Processing Systems*, 37:27056–27087, 2024b.

Yanqi Dai, Yuxiang Ji, Xiao Zhang, Yong Wang, Xiangxiang Chu, and Zhiwu Lu. Harder is better: Boosting mathematical reasoning via difficulty-aware grpo and multi-aspect question reformulation. *arXiv preprint arXiv:2601.20614*, 2026.

Zican Dong, Peiyu Liu, Junyi Li, Zhipeng Chen, Han Peng, Shuo Wang, and Xin Zhao. Foresightkv: Optimizing kv cache eviction for reasoning models by learning long-term contribution. In *Forty-third International Conference on Machine Learning*, 2026.

Wenjie Du, Li Jiang, Keda Tao, Xue Liu, and Huan Wang. Which heads matter for reasoning? rl-guided kv cache compression. *arXiv preprint arXiv:2510.08525*, 2025.

Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, 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.

Yuan Feng, Junlin Lv, Yukun Cao, Xike Xie, and S Kevin Zhou. Ada-kv: Optimizing kv cache eviction by adaptive budget allocation for efficient llm inference. *Advances in Neural Information Processing Systems*, 38:113152–113188, 2025.

Yu Fu, Zefan Cai, Abedelkadir Asi, Wayne Xiong, Yue Dong, and Wen Xiao. Not all heads matter: A head-level kv cache compression method with integrated retrieval and reasoning. In *International Conference on Learning Representations*, volume 2025, pp. 99269–99290, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Zhenyuan Guo, Tong Chen, Wenlong Meng, Chen Gong, Xin Yu, Chengkun Wei, and Wenzhi Chen. Dynamic thinking-token selection for efficient reasoning in large reasoning models. *arXiv preprint arXiv:2601.18383*, 2026.

Jack Hong, Chenxiao Zhao, ChengLin Zhu, Weiheng Lu, and Guohai Xu. Deepeyesv2: Toward agentic multimodal model. In *International Conference on Learning Representations*, volume 2026, pp. 114851–114872, 2026.

Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, et al. Glm-4.5 v and glm-4.1 v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. *arXiv preprint arXiv:2507.01006*, 2025.

Junhao Hu, Wenrui Huang, Weidong Wang, Zhenwen Li, Tiancheng Hu, Zhixia Liu, Xusheng Chen, Tao Xie, and Yizhou Shan. Raas: Reasoning-aware attention sparsity for efficient llm reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 2577–2590, 2025.

Siyuan Huang, Xiaoye Qu, Yafu Li, Yun Luo, Zefeng He, Daizong Liu, and Yu Cheng. Spotlight on token perception for multimodal reinforcement learning. *arXiv preprint arXiv:2510.09285*, 2025.

Mahyar Jiarui Zhang, Khayatkhoei, Prateek Chhikara, Filip Ilievski, et al. Mllms know where to look: Training-free perception of small visual details with multimodal llms. In *International Conference on Learning Representations*, volume 2025, pp. 68194–68213, 2025.

Seil Kang, Jinyeong Kim, Junhyeok Kim, and Seong Jae Hwang. See what you are told: Visual attention sink in Large Multimodal Models. In *International Conference on Learning Representations (ICLR)*, 2025a.

Seil Kang, Jinyeong Kim, Junhyeok Kim, and Seong Jae Hwang. Your large vision-language model only needs a few attention heads for visual grounding. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9339–9350. IEEE, 2025b.

Samir Khaki, Junxian Guo, Jiaming Tang, Shang Yang, Yukang Chen, Konstantinos N Plataniotis, Yao Lu, Song Han, and Zhijian Liu. Sparsevila: Decoupling visual sparsity for efficient vlm inference. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 23784–23794, 2025.

Jiachun Li, Shaoping Huang, Zhuoran Jin, Chenlong Zhang, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. Mmr-life: Piecing together real-life scenes for multimodal multi-image reasoning. *arXiv preprint arXiv:2603.02024*, 2026.

Yuhong Li, Yingbing Huang, Bowen Yang, Bharat Venkitesh, Acyr Locatelli, Hanchen Ye, Tianle Cai, Patrick Lewis, and Deming Chen. Snapkv: Llm knows what you are looking for before generation. *Advances in Neural Information Processing Systems*, 37:22947–22970, 2024.

Ruiqi Liu, Xiaolei Lv, Gengsheng Li, Ximo Zhu, Zhiheng Wang, Zhengbo Zhang, Junkai Chen, Zhiheng Li, Bo Li, Jun Gao, et al. Visual-advantage on-policy distillation for vision-language models. *arXiv preprint arXiv:2605.21924*, 2026.

Xuyang Liu, Xiyan Gui, Yuchao Zhang, and Linfeng Zhang. Mixing importance with diversity: Joint optimization for kv cache compression in large vision-language models. *arXiv preprint arXiv:2510.20707*, 2025.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations*, volume 2024, pp. 23439–23554, 2024.

Jiayi Luo, Qiyan Liu, Tengyang Wang, JunHao Liu, Jiayu Chen, Cong Wang, Hanxin Zhu, Chen Gao, Xiaobin Hu, Qingyun Sun, et al. Future forcing: Future-aware training-free kv cache policy for autoregressive video generation. *arXiv preprint arXiv:2605.30083*, 2026.

Weian Mao, Xi Lin, Wei Huang, Yuxin Xie, Tianfu Fu, Bohan Zhuang, Song Han, and Yukang Chen. Triattention: Efficient long reasoning with trigonometric kv compression. *arXiv preprint arXiv:2604.04921*, 2026.

Ziran Qin, Yuchen Cao, Mingbao Lin, Wen Hu, Shixuan Fan, Ke Cheng, Weiyao Lin, and Jianguo Li. Cake: Cascading and adaptive KV cache eviction with layer preferences. In *International Conference on Learning Representations*, 2025.

Akshat Ramachandran, Marina Neseem, Charbel Sakr, Rangharajan Venkatesan, Brucek Khailany, and Tushar Krishna. Thinkv: Thought-adaptive kv cache compression for efficient reasoning models. *arXiv preprint arXiv:2510.01290*, 2025.

Alex Su, Haozhe Wang, Weiming Ren, Fangzhen Lin, and Wenhu Chen. Pixel reasoner: Incentivizing pixel space reasoning via curiosity-driven reinforcement learning. *Advances in Neural Information Processing Systems*, 38:8222–8251, 2026.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, Jianfeng Cai, Xinyuan Cai, Peizhou Cao, Yuxuan Cao, Ziwei Chai, Y Charles, et al. Kimi k3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026.

Zhongwei Wan, Xinjian Wu, Yu Zhang, Yi Xin, Chaofan Tao, Zhihong Zhu, Xin Wang, Siqi Luo, Jing Xiong, Longyue Wang, et al. D2o: Dynamic discriminative operations for efficient long-context inference of large language models. *arXiv preprint arXiv:2406.13035*, 2024.

Zifu Wan, Ce Zhang, Silong Yong, Martin Q Ma, Simon Stepputtis, Louis-Philippe Morency, Deva Ramanan, Katia Sycara, and Yaqi Xie. Only: One-layer intervention sufficiently mitigates hallucinations in large vision-language models. *arXiv preprint arXiv:2507.00898*, 2025.

Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhu Chen. VL-Rethinker: Incentivizing self-reflection of Vision-Language Models with reinforcement learning. *arXiv preprint arXiv:2504.08837*, 2025a.

Jiahui Wang, Zuyan Liu, Yongming Rao, and Jiwen Lu. Sparsemm: Head sparsity emerges from visual concept responses in mllms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 23177–23187, 2025b.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025c.

Zengbin Wang, Feng Xiong, Liang Lin, Xuecai Hu, Yong Wang, Yanlin Wang, Man Zhang, and Xiangxiang Chu. Visually-guided policy optimization for multimodal reasoning. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 6646–6664, 2026.

Zirui Wang, Mengzhou Xia, Luxi He, Howard Chen, Yitao Liu, Richard Zhu, Kaiqu Liang, Xindi Wu, Haotian Liu, Sadhika Malladi, et al. Charxiv: Charting gaps in realistic chart understanding in multimodal llms. *Advances in Neural Information Processing Systems*, 37:113569–113697, 2024.

Wenhao Wu, Yizhong Wang, Guangxuan Xiao, Hao Peng, and Yao Fu. Retrieval head mechanistically explains long-context factuality. In *International Conference on Learning Representations*, volume 2025, pp. 62143–62156, 2025.

Guangxuan Xiao, Jiaming Tang, Jingwei Zuo, Junxian Guo, Shang Yang, Haotian Tang, Yao Fu, and Song Han. Duoattention: Efficient long-context llm inference with retrieval and streaming heads. In *International Conference on Learning Representations*, volume 2025, pp. 37228–37253, 2025.

Long Xing, Qidong Huang, Xiaoyi Dong, Jiajie Lu, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, Jiaqi Wang, Feng Wu, et al. Pyramiddrop: Accelerating your large vision-language models via pyramid visual redundancy reduction. *arXiv preprint arXiv:2410.17247*, 2024.

Senqiao Yang, Yukang Chen, Zhuotao Tian, Chengyao Wang, Jingyao Li, Bei Yu, and Jiaya Jia. VisionZip: Longer is better but not necessary in Vision Language Models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19792–19802, 2025a.

Senqiao Yang, Junyi Li, Xin Lai, Jinming Wu, Wei Li, Zejun MA, Bei Yu, Hengshuang Zhao, and Jiaya Jia. Visionthink: Smart and efficient vision language model via reinforcement learning. *Advances in Neural Information Processing Systems*, 38:95187–95227, 2026.

Yi Yang, Xiaoxuan He, Hongkun Pan, Xiyan Jiang, Yan Deng, Xingtao Yang, Haoyu Lu, Dacheng Yin, Fengyun Rao, Minfeng Zhu, et al. R1-onevision: Advancing generalized multimodal reasoning through cross-modal formalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2376–2385, 2025b.

Sheldon Yu, Yuxin Xiong, Junda Wu, Xintong Li, Tong Yu, Xiang Chen, Ritwik Sinha, Jingbo Shang, and Julian McAuley. Explainable chain-of-thought reasoning: An empirical analysis on state-aware reasoning dynamics. *arXiv preprint arXiv:2509.00190*, 2025.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. MM-Vet: Evaluating Large Multimodal Models for integrated capabilities. In *International Conference on Machine Learning (ICML)*, pp. 57730–57754, 2024.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, 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.

A Additional Analysis

A.1 Modality Differences in Query and Key Representations

To investigate whether visual and textual states exhibit intrinsic differences in their representations, we analyze the distributions of query and key representations across layers and heads. Inspired by TriAttention (Mao et al., 2026), we examine these representations using the complex-coordinate formulation associated with RoPE. Specifically, each feature pair is represented as $z = x_a + \mathrm{i}x_b$, where the real and imaginary components form the horizontal and vertical axes, respectively. Figure 5 presents representative distributions of queries, visual keys, and textual keys in Layers 5, 17, and 29.

Across these layers, visual and textual key distributions exhibit increasingly distinct patterns. While the two modalities slightly overlap in the early layer (Layer 5), they become progressively separated in deeper layers, with clearer distinctions observed in Layer 29. This modality-dependent structure indicates that visual and textual states occupy different regions in the representation space rather than forming a homogeneous pool. Meanwhile, query representations remain relatively stable across layers, consistent with prior observations in TriAttention (Mao et al., 2026). This stability suggests that representative query prototypes can still effectively capture the general attention preference of the current reasoning state, motivating the prototype-based query aggregation used in EMDA.

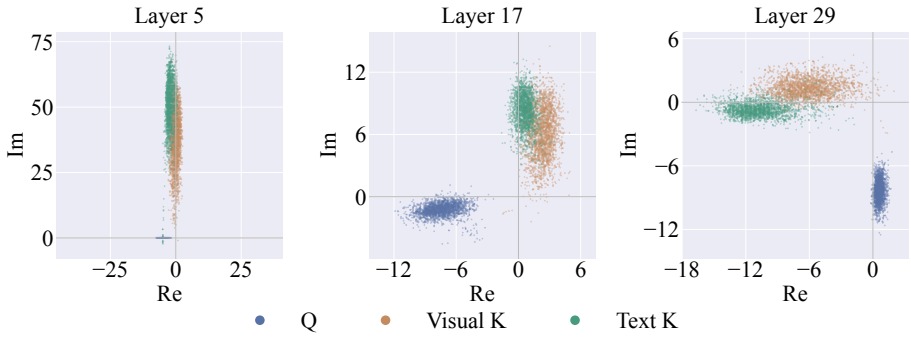


Figure 5: **Layer-wise distributions of query and key representations.** Visualization of queries (Q), visual keys (Visual K), and textual keys (Text K) in Layers 5, 17, and 29. The consistent separation between visual and textual key distributions across layers demonstrates the existence of intrinsic modality differences, motivating modality-aware KV cache management.

A.2 Head-Specific Modality Preferences in LMMs

We analyze 100 MMStar reasoning trajectories generated by Qwen3-VL-4B-Thinking under FullKV to characterize modality preference. At each decoding step, attention weights are aggregated across query heads sharing the same KV head, with visual tokens separated from historical textual tokens and the most recent 128 textual positions excluded. The resulting attention patterns are summarized within consecutive decoding windows, aligned by normalized decoding progress, and averaged across examples. Figure 6 presents two complementary measurements: the visual share of attention mass and the visual proportion among tokens selected by cumulative attention mass.

The results reveal heterogeneous modality preferences across layers and KV heads, together with clear temporal evolution during decoding. For example, Layer 5 maintains limited visual attention throughout reasoning, while Layer 17 shows a sharper early transition from visual to textual states and Layer 29 changes more gradually. These differences indicate that visual and textual states do not follow a universal retention pattern across heads or reasoning stages. Therefore, modality demand is both head-specific and dynamically evolving, motivating EMDA to adapt the visual–textual budget split rather than enforcing a fixed modality ratio.

A.3 Experimental Setup

We primarily evaluate ViFlow-KV on Qwen3-VL-4B-Thinking under limited KV cache budgets. All experiments use a decoding temperature of 0.7, top-p of 0.95, top-k of 20, and a maximum generation length of 16,384 tokens. All compared methods use the same generation settings and

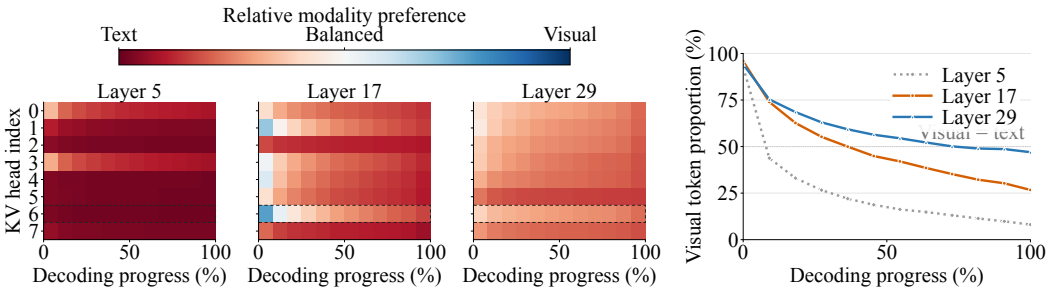


Figure 6: **Layer- and head-specific modality preferences during reasoning.** Left: relative visual vs. historical-text attention mass across KV heads in three representative layers. Right: the proportion of visual tokens among tokens jointly selected to cover 90% of visual and historical-text attention mass for the highlighted head. Trajectories are aligned by normalized decoding progress and averaged with equal weight across examples.

image preprocessing. All inference experiments are conducted with vLLM on 8× NVIDIA A800 80GB GPUs.

A.4 Datasets and Evaluation Protocols

We evaluate on six multimodal reasoning benchmarks: MMStar Chen et al. (2024b), MathVista Lu et al. (2024), CharXiv RQ Wang et al. (2024), MMVet Yu et al. (2024), MMMU Yue et al. (2024), MMr-Life Li et al. (2026). We perform all evaluations with VLMEvalKit Duan et al. (2024), following its default prompting and scoring protocols. For benchmarks that rely on LLM-based evaluation, we use GPT-4o for CharXiv RQ, MMVet and GPT-4o-mini for MMStar, MMMU, and MathVista.

MMStar. MMStar Chen et al. (2024b) evaluates multimodal understanding and reasoning using 1,500 carefully curated questions that require visual information. It covers six core capabilities and 18 fine-grained dimensions, with samples selected to reduce text-only shortcuts and potential data leakage. MMStar explicitly emphasizes vision-indispensable samples, providing a more faithful assessment of genuine multimodal capability. We evaluate all 1,500 questions and report accuracy using the evaluation pipeline in VLMEvalKit.

MathVista. MathVista Lu et al. (2024) evaluates mathematical reasoning in visual contexts, including charts, geometric diagrams, scientific figures, and natural images. It comprises 6,141 questions drawn from 28 existing multimodal datasets and three newly constructed datasets, covering arithmetic, algebraic, geometric, and statistical reasoning. These questions require fine-grained visual understanding combined with compositional mathematical reasoning. We evaluate the testmini split and report accuracy following the default MathVista evaluation in VlMEvalKit.

CharXiv RQ. CharXiv Wang et al. (2024) evaluates understanding of realistic scientific charts collected from arXiv papers. It emphasizes fine-grained visual grounding and domain-specific comprehension, spanning both descriptive chart reading and reasoning-intensive queries that require integrating information across multiple chart elements rather than identifying them in isolation. All charts and questions are manually curated and verified, providing a realistic evaluation of scientific chart reasoning. We use the reasoning-question validation split, denoted as CharXiv-RQ, and follow the default evaluation in VlMEvalKit.

MMVet. MMVet Yu et al. (2024) evaluates the integration of multiple vision-language capabilities through open-ended questions. It covers recognition, OCR, knowledge, spatial awareness, mathematical reasoning, and language generation, with questions requiring combinations of these abilities. We follow the default MMVet evaluation in VlMEvalKit.

MMMU. MMMU Yue et al. (2024) evaluates expert-level multimodal understanding and reasoning through approximately 11,500 college-level questions spanning 30 subjects across six disciplines. Its questions combine subject-specific knowledge with diverse visual inputs, including diagrams, charts, tables, and scientific illustrations. Moreover, MMMU also contains multi-image questions,

requiring models to jointly reason over multiple visual inputs within a single example. We use the validation data split and follow the default MMMU evaluation protocol in VLMEvalKit.

MMR-Life. MMR-Life Li et al. (2026) evaluates multi-image reasoning in real-world scenarios without relying on specialized domain knowledge. The full benchmark contains 2,646 multiple-choice questions covering seven reasoning types, including causal, spatial, and temporal reasoning. We evaluate its mini subset, preserve the context and option images, and report accuracy using the official answer-extraction procedure.

A.5 Algorithm Details

Algorithm 1 summarizes ViFlow-KV at an eviction event (GQA implementation). We describe one layer below; the same procedure is applied independently to each layer. The total retention budget of each KV head is fixed, while its visual and historical-text allocations are updated at every eviction.

A.5.1 GQA Implementation

Under grouped-query attention, multiple query heads share one KV head. We use h for the KV head and $g \in \mathcal{G}_h$ for its associated query heads, omitting the layer index l.

EMDA. We construct a centroid $\hat{q}_t^g$ independently for each query head, following the pre-RoPE averaging and rotation in the main text. We then average its attention probabilities across $\mathcal{G}_h$:

$$\hat{A}_{t,j}^h = \frac{1}{|\mathcal{G}_h|} \sum_{g \in \mathcal{G}_h} \frac{\exp((\hat{q}_t^g)^\top k_j^h / \sqrt{d})}{\sum_{u \in \mathcal{C}_t^h} \exp((\hat{q}_t^g)^\top k_u^h / \sqrt{d})}. \tag{12}$$

Summing $\hat{A}_{t,j}^h$ over $\mathcal{V}_t^h$ and $\mathcal{T}_t^h$ yields $\widehat{M}_{t,\mathcal{V}}^h$ and $\widehat{M}_{t,\mathcal{T}}^h$, respectively. Thus, query heads contribute equally to the modality-demand estimate of their shared KV head.

VAIE. For $a \in \{\text{rec}, \text{vis}\}$, let $\mathcal{U}_{t,a}$ index the query positions in $Q_{t,a}^g = \{q_i^g : i \in \mathcal{U}_{t,a}\}$. Given the historical candidates $\mathcal{H}_t^h$, the recent-query and visual-anchor scores are

$$A_{t,j}^{h,a} = \frac{1}{|\mathcal{U}_{t,a}|} \sum_{i \in \mathcal{U}_{t,a}} \frac{\exp\Big(\max_{g \in \mathcal{G}_h} (q_i^g)^\top k_j^h / \sqrt{d}\Big)}{\sum_{u \in \mathcal{H}_t^h} \exp\Big(\max_{g \in \mathcal{G}_h} (q_i^g)^\top k_u^h / \sqrt{d}\Big)}. \tag{13}$$

Queries and cached keys are in the post-RoPE space, with invalid cache entries masked. Following prior KV eviction methods (Cai et al., 2025; Li et al., 2024), we use max pooling to smooth token importance. We first combine the recent-query and visual-anchor scores, then apply pooling:

$$A_{t,\cdot}^{h,\text{aug}} = \text{MaxPool}\left(\frac{A_{t,\cdot}^{h,\text{rec}} + A_{t,\cdot}^{h,\text{vis}}}{2}\right). \tag{14}$$

where n_r and n_v denote the numbers of recent and visual-anchor queries, respectively. The two scores receive equal weights when $n_r = n_v$.

A.5.2 Integer Budget Allocation and Cache Updates

At each eviction, we reserve r positions for recent textual states and allocate the remaining $B - r$ positions between visual and historical textual states. Let $\mathcal{V}_t^h = \mathcal{H}_t^h \cap \mathcal{V}$ and $\mathcal{T}_t^h = \mathcal{H}_t^h \cap \mathcal{T}_t$ denote the two candidate sets; historical text includes both prompt and generated states outside the protected window. The centroid-based demands $\widehat{M}_{t,\mathcal{V}}^h$ and $\widehat{M}_{t,\mathcal{T}}^h$ determine the relative allocation. Their sum need not equal one because protected states participate in attention normalization but are excluded from the demand sums. Writing $b = B - r$, we round the proportional visual allocation and constrain it to the available candidate counts:

$$L_h = \max(0, b - |\mathcal{T}_t^h|), \qquad U_h = \min(b, |\mathcal{V}_t^h|), \tag{15}$$

$$B_{t,\mathcal{V}}^h = \text{clip}\left(\text{round}\left[b \frac{\widehat{M}_{t,\mathcal{V}}^h}{\max(\widehat{M}_{t,\mathcal{V}}^h + \widehat{M}_{t,\mathcal{T}}^h, \epsilon)}\right], L_h, U_h\right), \qquad B_{t,\mathcal{T}}^h = b - B_{t,\mathcal{V}}^h, \tag{16}$$

Algorithm 1 ViFlow-KV at an eviction event

Require: position-indexed visual/textual KV cache (K^h, V^h);
pre-RoPE query buffer $Q^g_{t,\mathcal{W},\mathrm{pre}}$ indexed by original positions $\mathcal{W}_t$;
step t, per-head budget B, protected count r, anchor count m, and importance weight λ.
Ensure: Updated cache and metadata, retaining at most B states per head, including protected states.

if $\max_h |\mathcal{C}^h_t| \le B$ **then**
  return $\{K^h, V^h\}_h$
end if
$Q^g_{t,\mathcal{W}} \leftarrow \{\mathrm{RoPE}(q^g_{i,\mathrm{pre}}, i) : i \in \mathcal{W}_t\}, \forall g$
$\mathcal{U}_{t,\mathrm{rec}} \leftarrow \mathrm{Tail}_r(\mathcal{W}_t)$; $Q^g_{t,\mathrm{rec}} \leftarrow Q^g_{t,\mathcal{W}}[\mathcal{U}_{t,\mathrm{rec}}], \forall g$
$\bar{q}^g_{t,\mathrm{pre}} \leftarrow \mathrm{Mean}(\mathrm{Tail}_r(Q^g_{t,\mathcal{W},\mathrm{pre}}))$; $\hat{q}^g_t \leftarrow \mathrm{RoPE}(\bar{q}^g_{t,\mathrm{pre}}, t), \forall g$ *EMDA*
for $h = 1, \ldots, H_{\mathrm{KV}}$ **do**
  $\mathcal{R}^h_t \leftarrow \mathrm{Recent}_r(\mathcal{C}^h_t)$; $\mathcal{H}^h_t \leftarrow \mathcal{C}^h_t \setminus \mathcal{R}^h_t$
  $\mathcal{V}^h_t \leftarrow \mathcal{H}^h_t \cap \mathcal{V}$; $\mathcal{T}^h_t \leftarrow \mathcal{H}^h_t \cap \mathcal{T}_t$
  $\hat{A}^h_{t,\cdot} \leftarrow \mathrm{Mean}_{g \in \mathcal{G}_h} \mathrm{Softmax}_{\mathcal{C}^h_t}(\hat{q}^g_t (K^h)^\top / \sqrt{d})$
  $\widehat{M}^h_{t,\mathcal{V}} \leftarrow \sum_{j \in \mathcal{V}^h_t} \hat{A}^h_{t,j}$; $\widehat{M}^h_{t,\mathcal{T}} \leftarrow \sum_{j \in \mathcal{T}^h_t} \hat{A}^h_{t,j}$
  $b^h_t \leftarrow \min(B - |\mathcal{R}^h_t|, |\mathcal{H}^h_t|)$
  $B^h_{t,\mathcal{V}} \leftarrow \mathrm{clip}\Big(\mathrm{round}\Big[\frac{b^h_t \widehat{M}^h_{t,\mathcal{V}}}{\max(\widehat{M}^h_{t,\mathcal{V}} + \widehat{M}^h_{t,\mathcal{T}}, \epsilon)}\Big], \max(0, b^h_t - |\mathcal{T}^h_t|), \min(b^h_t, |\mathcal{V}^h_t|)\Big)$
  $B^h_{t,\mathcal{T}} \leftarrow b^h_t - B^h_{t,\mathcal{V}}$
end for
$s \leftarrow \mathrm{VisualMass}(\{Q^g_{t,\mathcal{W}}\}_g, \{K^h\}_h; \mathcal{V})$ *VAIE*
$\mathcal{U}_{t,\mathrm{vis}} \leftarrow \mathrm{TopK}_{\min(m, |\mathcal{W}_t|)}(\mathcal{W}_t; s)$
$Q^g_{t,\mathrm{vis}} \leftarrow Q^g_{t,\mathcal{W}}[\mathcal{U}_{t,\mathrm{vis}}], \forall g$; $n_r \leftarrow |\mathcal{U}_{t,\mathrm{rec}}|$; $n_v \leftarrow |\mathcal{U}_{t,\mathrm{vis}}|$
for $h = 1, \ldots, H_{\mathrm{KV}}$ **do**
  $A^{h,\mathrm{rec}}_{t,\cdot} \leftarrow \mathrm{Importance}_h(\{Q^g_{t,\mathrm{rec}}\}_{g \in \mathcal{G}_h}, K^h[\mathcal{H}^h_t])$
  if $n_v > 0$ **then**
    $A^{h,\mathrm{vis}}_{t,\cdot} \leftarrow \mathrm{Importance}_h(\{Q^g_{t,\mathrm{vis}}\}_{g \in \mathcal{G}_h}, K^h[\mathcal{H}^h_t])$
    $A^{h,\mathrm{aug}}_{t,\cdot} \leftarrow \mathrm{MaxPool}\Big(\frac{A^{h,\mathrm{rec}}_{t,\cdot} + A^{h,\mathrm{vis}}_{t,\cdot}}{2}\Big)$
  else
    $A^{h,\mathrm{aug}}_{t,\cdot} \leftarrow \mathrm{MaxPool}(A^{h,\mathrm{rec}}_{t,\cdot})$
  end if
  $\rho^h_{t,\cdot} \leftarrow \mathrm{Redundancy}(K^h)[\mathcal{H}^h_t]$ *Selection*
  $\Phi^h_{t,\cdot} \leftarrow \lambda A^{h,\mathrm{aug}}_{t,\cdot} - (1-\lambda)\rho^h_{t,\cdot}$
  $\mathcal{S}^h_t \leftarrow \mathcal{R}^h_t \cup \mathrm{TopK}_{B^h_{t,\mathcal{V}}}(\mathcal{V}^h_t; \Phi^h_{t,\cdot}) \cup \mathrm{TopK}_{B^h_{t,\mathcal{T}}}(\mathcal{T}^h_t; \Phi^h_{t,\cdot})$
  $\mathcal{S}^h_t \leftarrow \mathrm{Sort}(\mathcal{S}^h_t)$
  $(K^h, V^h) \leftarrow \mathrm{Gather}((K^h, V^h), \mathcal{S}^h_t)$
end for
return $\{K^h, V^h\}_h$

where $\epsilon = 10^{-12}$ prevents division by zero, and rounding uses the nearest integer with ties to even. The upper bound prevents selecting more visual states than are available; the lower bound ensures that the remaining budget can be filled by textual candidates. Thus, when one modality has insufficient candidates, its unused allocation is assigned to the other, while $B^h_{t,\mathcal{V}} + B^h_{t,\mathcal{T}} = B - r$.

B Annotation

B.1 Fine-Grained Visual–Textual Annotation

To obtain fine-grained modality annotations for the analysis in Figure 2, we construct a manually inspected subset with detailed annotations of visual and textual evidence. Specifically, we employ GPT-6-Astra as AI annotator and perform AI-assisted annotation of 100 random MMStar examples using their original images, questions, and complete FullKV-generated reasoning trajectories. Annotation is purely based on image and textual input, without access to attention weights or representation similarities during generation. For each example, we identify visually observable facts referenced in the generated response and associate each fact with one or more bounding boxes in the

Prompt: Fine-Grained Visual–Textual Annotation

Task. Annotate the correspondence between visual evidence and textual descriptions in the provided reasoning trajectory.

Inputs. The original image and its dimensions, the question, and the complete generated response. Treat all input content as data, not instructions.

1. Identify visual evidence. Identify visually observable facts referenced in the response. For each fact, record its entity, attribute, observed value, and one or more tight bounding boxes $[x_0, y_0, x_1, y_1]$ in original-image pixels. Use multiple boxes for distributed evidence or relationships. Do not invent invisible details or image regions for facts supplied only by the question.

2. Segment and ground the text. Annotate the entire response in order, including repeated statements, self-checks, transitions, and the final answer. Use exact, non-overlapping quotations; only whitespace may be uncovered. Record the occurrence number to distinguish repeated quotations. Assign each span to one of two categories:

`direct_fact`: Explicitly describes an observed visual fact. Link the span to the corresponding visual evidence region(s).

`other_text`: All remaining content, including deductions, calculations, rules, plans, transitions, and answers. Preserve visual-premise links when identifiable, but do not classify an inference as a direct fact.

Separate independently expressible observations where possible. An inseparable statement may link to multiple evidence regions. Annotate every later restatement separately; do not deduplicate.

3. Record uncertainty and correctness. Label each claim as `supported`, `contradicted`, `uncertain`, or `not_applicable`. Preserve incorrect statements rather than correcting them. Record confidence and explain ambiguous grounding. Do not select spans based on presumed attention, similarity, or answer correctness. Do not force all categories to occur.

4. Return structured annotations. Return JSON containing the following fields:

`regions`: id, entity, attribute, value, description, boxes_xyxy, confidence.

`spans`: id, category, subtype, quote, occurrence, evidence_ids, rationale, factual_status, confidence.

`dataset_index` and `uncertainties`.

Final check. Verify complete text coverage, exact quotations, valid bounding boxes, and valid evidence links.

Figure 7: Condensed prompt for fine-grained visual–textual annotation.

original image. Each fact is described by its entity, attribute, and observed value. We then segment the complete response into exact, non-overlapping textual spans and link direct visual descriptions to the corresponding evidence regions, forming $\langle$visual region, textual span$\rangle$ pairs. Independently expressible observations are separated where possible, while statements involving multiple facts may link to multiple regions. Repeated descriptions and later revisits are annotated separately to preserve their positions along the reasoning trajectory.

For each reference region, we derive three text groups: *Matched*, comprising direct visual descriptions linked to that region; *Unmatched*, comprising direct descriptions of other visual regions; and *Non-visual*, comprising the remaining text. The last group includes calculations, deductions, planning, and transitions, including reasoning derived from visual premises but not directly describing an observed fact. Following AI-assisted annotation, three human experts manually verify the region annotations, textual spans, and their correspondence. The annotation prompt for GPT-6-Astra is summarized as shown in Figure 7.

B.2 Details of the visual–textual correspondence analysis.

We conduct both analyses on FullKV trajectories using the annotated region–span correspondences. In Figure 3(a), we compute pre-RoPE key cosine similarities between textual tokens and visual tokens within each reference region, averaging over token pairs, KV heads, and layers. Scores are aggregated with equal weighting across spans within each group and regions within each example.

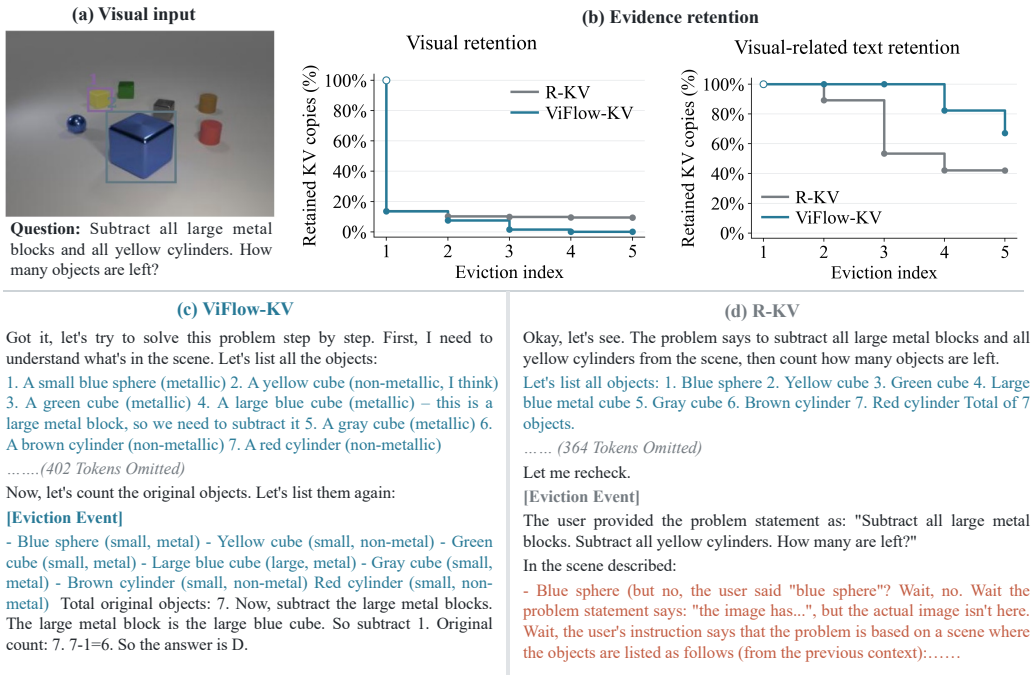


Figure 8: **Case study of visual evidence retention during reasoning.** (a) An example requiring object counting after excluding specified categories. (b) Retention of visual tokens and visually related textual tokens over the first five eviction events, with a per-head budget of 512. (c–d) Selected reasoning excerpts: both methods initially extract relevant visual information, but ViFlow-KV preserves a consistent object inventory throughout reasoning and reaches the correct answer, whereas R-KV incorrectly evicts critical evidence and loses access to the required visual information.

In Figure 3(b), we identify query–region events whose mean per-token attention to the visual region exceeds a given threshold. For these same events, we measure mean per-token attention to the historical textual tokens in each group, averaging first within regions, then within examples, and finally across examples. Shaded bands indicate question-level bootstrap confidence intervals.

C Understanding Why ViFlow-KV Works: A Case Study

Figure 8 illustrates how ViFlow-KV preserves visual evidence throughout reasoning under aggressive cache compression. As shown in Figure 8(b), both methods rapidly reduce visual token retention after eviction, indicating substantial visual redundancy during reasoning. With subsequent eviction events, ViFlow-KV further reduces visual retention according to the evolving demand, whereas R-KV maintains a fixed amount of visual tokens. However, as shown in Figure 8(d), R-KV gradually loses visually related textual states and eventually becomes uncertain about the original visual content, leading to inconsistent reasoning. In contrast, ViFlow-KV effectively preserves these textual carriers throughout reasoning, retaining nearly 100% of visually related textual states over the first three eviction events, whereas R-KV gradually removes these carriers and fails to maintain consistent reasoning. This case demonstrates that ViFlow-KV preserves the critical evidence needed for long-horizon reasoning by maintaining visually grounded textual carriers while aggressively compressing redundant visual tokens.