

Article

Cross-Modal Variance-Aware KV Cache Optimization for Efficient Multimodal Long-Context Inference

Shenglong Liu ¹, Yanli Lv ¹, Siyao An ^{1,*}, Chenghuan Yu ^{2,3} and Yiwei Ru ²

¹ Network and Data Security Department, State Grid Information & Telecommunication Center (Big Data Center), Beijing 100761, China; shenglong-liu@sgcc.com.cn (S.L.); yanli-lv@sgcc.com.cn (Y.L.)

² Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China; chenghuan.yu@rwth-aachen.de (C.Y.); yiwei.ru@cripac.ia.ac.cn (Y.R.)

³ RWTH International Academy, RWTH Aachen University, 52062 Aachen, Germany

* Correspondence: siyao-an@sgcc.com.cn

Abstract

Multimodal large language models (MLLMs) face substantial memory bottlenecks when processing long visual contexts, such as videos and high-resolution images. Existing methods that allocate visual KV cache budgets using cross-modal attention entropy mainly estimate the distributional breadth of text–vision interaction and may overlook how visual relevance varies across query positions in the encoded multimodal context. We propose a cross-modal query-position variance-aware KV cache optimization method for efficient multimodal long-context inference. The proposed method combines cross-modal attention entropy with prefill-stage query-position variance computed from cross-modal attention to estimate layer-wise visual KV cache preferences. Based on this preference score, visual KV cache budgets are allocated across layers, and variance-aware token pruning is applied to retain high-importance KV states while directly evicting redundant visual tokens without feature merging. Experiments on the MileBench benchmark using LLaVA-v1.5-7B show that, under deterministic single-run evaluation and while retaining only 20% of the visual KV cache, the proposed method produces point-estimate performance close to the full-cache reference and higher point estimates on several fine-grained reasoning and retrieval subtasks. Additional representative-subtask evaluations under different visual cache budgets and on InternVL2.5-8B further provide preliminary point-estimate evidence that the proposed allocation signal is not restricted to a single cache ratio or backbone. System profiling further shows that the 20% cache setting reduces measured KV cache GPU memory from 1.28 GiB to 0.26 GiB and decoding latency from 100.28 ms/token to 92.85 ms/token. These results suggest that cross-modal query-position variance may help preserve sparse, query-dependent visual cues under low-cache-budget multimodal inference.

Keywords: Multimodal Large Language Models; KV cache optimization; cross-modal query-position variance; long-context inference; multimodal reasoning



Academic Editor: Christos J. Bouras

Received: 26 May 2026

Revised: 3 July 2026

Accepted: 6 July 2026

Published: 21 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Over the past several years, the Transformer [1] architecture has become a foundational paradigm in natural language processing. As model parameters increase and context windows expand to hundreds of thousands of tokens, both academia and industry face growing computational and memory challenges. During autoregressive decoding, the model must attend to the hidden states of all preceding tokens when generating each

new token. To avoid repeated computation over the full context, modern inference systems cache the key and value states of previous tokens, a mechanism known as the KV cache [2].

However, the KV cache also introduces a major memory bottleneck. Unlike model weights, which occupy a relatively fixed amount of memory during inference, KV cache storage grows linearly with sequence length and further increases with batch size. As multimodal large language models are increasingly used for long-video understanding, multi-image reasoning, and high-resolution visual perception, this growth becomes particularly severe.

The core of this challenge lies in the high dimensionality and redundancy of visual inputs. In text processing, a single token often corresponds to a word, subword, or character and usually carries relatively high semantic density. In contrast, for MLLMs to process an image, visual encoders such as CLIP-ViT-L/14 [3] partition the image into many patch tokens. A standard-resolution image is typically encoded into 576 visual tokens, and this number can increase rapidly for high-resolution images, multi-image inputs, or long videos. For example, a one-hour video sampled at 1 FPS may generate hundreds of thousands or even millions of visual tokens, depending on the visual encoder and tokenization strategy.

This asymmetry leads to an important observation: in MLLM inference, visual tokens can dominate the memory footprint, while their average semantic density is often lower than that of text tokens. For large models and long contexts, the KV cache alone can exceed the memory capacity of a single high-end GPU, depending on model size, numerical precision, sequence length, and batch size. This memory bottleneck not only restricts long-context multimodal processing but also limits system throughput. When available memory is constrained, inference systems may need to reduce batch size, which can lead to underutilized GPU computation.

Visual inputs also contain substantial redundancy. Spatial redundancy arises when neighboring image patches have similar visual features and provide limited additional semantic information. In video inputs, frame-level redundancy is also common because adjacent frames often differ only slightly and may correspond to the same semantic event or action. These observations suggest that a considerable portion of visual tokens may contribute little to a specific reasoning step, especially when the task depends on only a small subset of visual evidence.

Beyond efficiency, excessive visual redundancy may introduce distractors during long-context reasoning. A long video or multi-image input often contains many background patches or repeated visual patterns that are weakly related to the current question. Retaining all such tokens may dilute attention and make retrieval or fine-grained reasoning more difficult. Therefore, KV cache compression can be viewed not only as a memory-saving technique but also as a possible way to improve the signal-to-noise ratio of retained multimodal context. However, this paper does not claim verified adversarial robustness, hallucination suppression, or denial-of-service resistance. These issues are treated as deployment motivations and potential implications that require dedicated future evaluation.

Existing methods that allocate visual KV cache budgets using cross-modal attention entropy mainly estimate how broadly each layer participates in text–vision interaction. While this signal is useful, cross-modal attention entropy alone may overlook attention variation across query positions in the encoded multimodal context. In long-context multimodal inputs, some visual cues may receive weak average attention but become important for specific retrieval, comparison, or state-change queries. Such sparse and query-dependent visual cues are difficult to preserve under aggressive compression if allocation relies only on cross-modal attention entropy.

To address this limitation, this study proposes a cross-modal query-position variance-aware KV cache optimization method for efficient multimodal long-context inference. The proposed method combines cross-modal attention entropy with prefill-stage cross-

modal query-position variance to estimate layer-wise visual KV cache preferences. Cross-modal attention entropy captures the distributional breadth of text–vision interaction, whereas query-position variance measures how cross-modal attention to visual evidence fluctuates across query positions within the encoded context. Based on the resulting preference score, the method performs layer-wise adaptive budget allocation and applies variance-aware token pruning to evict redundant visual KV states without feature merging.

We summarize our contributions as follows:

- We extend cross-modal attention entropy-based visual KV cache allocation by incorporating prefill-stage cross-modal query-position variance, so that the layer preference score reflects both text–vision interaction breadth and query-position attention fluctuation.
- We design a visual-token-focused compression pipeline that combines modality-aware token slicing, layer-wise adaptive budget allocation, and variance-aware token pruning for redundant visual KV states.
- We provide preliminary empirical evidence on MileBench showing that, under deterministic single-run evaluation, the proposed method can retain only 20% of the visual KV cache while achieving point-estimate performance close to the full-cache reference. The results also show higher point estimates on several fine-grained reasoning and long-range retrieval subtasks, together with measured reductions in KV cache memory and decoding latency.

2. Related Work

KV cache optimization has become an important direction for improving the efficiency of long-context inference in Transformer-based models. Existing methods can be broadly divided into two lines: KV cache compression for language models and visual token or visual KV cache reduction for multimodal large language models. This section reviews these two lines of work and then discusses the gap addressed by the proposed method.

In addition to eviction-based methods, KIVI [4] studies low-bit KV cache quantization, while PagedAttention [5] improves serving-level KV cache memory management. These methods are complementary to our method, which focuses on modality-aware visual KV cache allocation rather than quantization or paging-based serving.

2.1. KV Cache Compression for Long-Context Language Models

During autoregressive decoding, the KV cache grows linearly with context length, making memory management a key bottleneck for long-context inference. Early methods mainly rely on simple retention rules or attention-based token selection. StreamingLLM [6] observes the attention sink phenomenon, where initial tokens receive consistently high attention, and proposes to keep these sink tokens together with a sliding window of recent tokens. This design enables stable streaming inference, but information located in the middle of long contexts may be lost.

H₂O [7] introduces a heavy-hitter-based eviction strategy. It retains tokens with high accumulated attention scores and evicts less-attended tokens from the KV cache. This strategy improves memory efficiency while preserving many important historical tokens. However, cumulative attention scores mainly reflect aggregated historical importance and may fail to preserve tokens with intermittent or context-dependent relevance.

More recent studies further consider the layer-wise structure of Transformer models. PyramidKV [8] shows that different layers have different cache requirements and proposes a pyramidal budget allocation strategy across layers. Larger budgets are assigned to lower layers to preserve detailed information, while higher layers are compressed more aggressively. CAKE [9] further introduces layer preferences based on attention statistics and

attention variability, showing that variance can help identify tokens that are not consistently highly attended but may still be important for later reasoning.

These studies indicate that KV cache compression should account for both token-level importance and layer-wise differences in attention behavior. However, most of these methods are designed for language-only inference and do not explicitly address modality imbalance or visual redundancy in multimodal contexts.

2.2. KV Cache and Visual Token Reduction for Multimodal Inference

Multimodal large language models introduce additional challenges because images and videos are usually represented by large numbers of visual tokens. Compared with text tokens, visual tokens often contain substantial spatial redundancy and, in video inputs, frame-level redundancy. Therefore, directly applying language-oriented KV cache compression methods may be suboptimal for multimodal inference.

FastCache [10] reduces the cost of multimodal inference by compressing visual KV cache representations with lightweight mechanisms. It shows that many visual tokens can be removed or merged with limited performance degradation. Look-M [11] focuses on efficient multimodal long-context inference and reduces redundant visual information through text-guided visual token selection. These methods demonstrate that visual redundancy can be exploited to improve inference efficiency.

MEDA [12] further proposes a cross-modal attention entropy-based KV cache allocation strategy for multimodal long-context inference. It uses cross-modal attention entropy to estimate the degree of text–vision interaction in each layer and allocates cache budgets accordingly. Layers with stronger and broader cross-modal interactions receive larger budgets, while layers with weaker interactions are compressed more aggressively. This cross-modal attention entropy-based allocation provides an effective way to estimate the distributional breadth of text–vision interaction.

However, cross-modal attention entropy mainly reflects the distributional breadth of text–vision interaction. In long-context multimodal reasoning, the relevance of visual cues may vary across query positions in the encoded multimodal context. Some visual tokens may receive weak average attention but become important for specific retrieval, comparison, or state-change queries. Such sparse and query-dependent cues are difficult to capture using cross-modal attention entropy alone.

2.3. MLLM Application Contexts

MLLMs have been applied beyond general visual question answering, including educational image-based vocabulary learning [13] and domain-specific visual reasoning in dental technology examinations [14]. These applications indicate that MLLMs are increasingly expected to process heterogeneous visual evidence in practical scenarios. Deployment in such scenarios also increases the demand for efficient long-context inference, especially when many images or long visual sequences must be processed. The proposed method is motivated by this broader need for efficient multimodal context handling, although the present work focuses on KV cache optimization rather than application-specific model evaluation.

2.4. Position of This Work

The proposed method is motivated by this limitation. Rather than replacing cross-modal attention entropy-based visual KV cache allocation, it extends this direction by introducing prefill-stage cross-modal query-position variance into the layer preference score. Cross-modal attention entropy is used to estimate the distributional breadth of text–vision interaction, while cross-modal query-position variance is used to measure attention fluctuation across query positions in the encoded multimodal context. By combining

these two signals, the proposed method aims to preserve visual tokens that are broadly relevant or important for specific query positions during long-context reasoning.

Compared with language-only KV cache compression methods, the proposed method explicitly considers the modality structure of multimodal inputs and applies compression mainly to visual tokens. Compared with existing multimodal methods, especially MEDA, the proposed method introduces prefill-stage cross-modal query-position variance into layer-wise visual KV cache budget allocation. This design is expected to be useful for tasks that require fine-grained visual comparison, state-change reasoning, or long-range retrieval, where key visual evidence may appear sparsely across the multimodal context.

Table 1 summarizes the conceptual differences between the proposed method and representative KV cache or visual token compression methods.

Table 1. Conceptual comparison between the proposed method and representative KV cache or visual token compression methods. The table summarizes methodological characteristics rather than experimental results.

Method	Multimodal-Aware	Allocation Signal	Layer-Wise Budget	Compression Operation
StreamingLLM	No	Attention sink	No	Token retention
H ₂ O	No	Accumulated attention	No	Token eviction
PyramidKV	No	Layer position	Yes	Budget allocation
CAKE	No	Attention statistics + variance	Yes	Token eviction
FastCache	Yes	Visual redundancy	Partial	KV compression
Look-M	Yes	Text-guided visual relevance	Partial	Token pruning
MEDA	Yes	Cross-modal attention entropy	Yes	Token merging
Ours	Yes	Entropy + query-position variance	Yes	Token pruning

Compared with CAKE, the proposed method does not directly transfer the variance metric to a language-only cache eviction setting. Instead, it adapts variance-based attention fluctuation modeling to cross-modal attention, where text and visual tokens have different semantic densities and cache priorities. Compared with MEDA, the proposed method retains the cross-modal attention entropy-based view of text–vision interaction but adds query-position variance to the layer allocation signal. In addition, while MEDA uses token merging after entropy-based budget allocation, the proposed method performs variance-aware token pruning by directly evicting low-scoring visual KV states without feature merging. This distinction preserves the original representations of retained KV states and makes the compression operation simple and reproducible.

3. Methodology

MEDA allocates multimodal KV cache budgets according to cross-modal attention entropy, which provides an effective estimate of the distributional breadth of text–vision interaction in each layer. However, cross-modal attention entropy mainly describes how broadly attention is distributed and does not explicitly capture how cross-modal attention varies across query positions in the encoded multimodal context. This limitation is important for long-context multimodal reasoning, where a visual cue may receive weak average attention but still become important for specific retrieval, comparison, or state-change queries.

To address this issue, we propose a cross-modal query-position variance-aware KV cache compression method. Different from MEDA, which uses cross-modal attention entropy for adaptive budget allocation followed by token merging, the proposed method combines cross-modal attention entropy with prefill-stage cross-modal query-position variance for layer-wise visual KV cache allocation and then applies variance-aware token

pruning. The resulting preference score is used to allocate visual KV cache budgets across layers, while the pruning module retains high-importance KV states and directly evicts low-scoring redundant visual tokens without feature merging. The overall framework is shown in Figure 1.

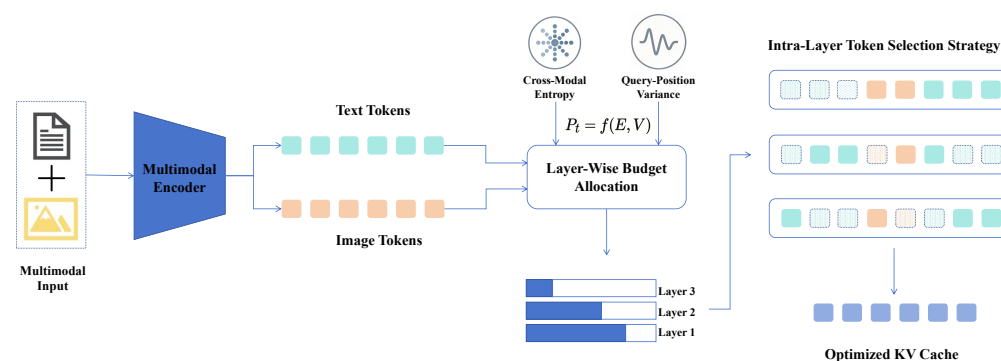


Figure 1. Overall framework of the proposed method, including modality-aware visual token slicing, layer-wise preference estimation, adaptive visual KV budget allocation, and variance-aware token pruning.

3.1. Notation and Computation Details

Let the multimodal input contain a text token index set T and a visual token index set V . For the l -th Transformer layer and the h -th attention head, the attention matrix is denoted as $A^{l,h} \in \mathbb{R}^{Q \times K}$, where Q is the number of query positions and $K = |T| + |V|$ is the total number of cached key positions. The multimodal token boundaries provided by image placeholders are used to separate text and visual tokens. The text-to-vision attention submatrix $A_{TV}^{l,h}$ is obtained by selecting text query positions and visual key positions, while the vision-to-text submatrix $A_{VT}^{l,h}$ is obtained by selecting visual query positions and text key positions.

For notational simplicity, $\bar{A}^l$ denotes the head-aggregated attention representation used for estimating cross-modal statistics in the l -th layer. The cross-modal attention entropy and cross-modal query-position variance are computed from the corresponding submatrices of $\bar{A}^l$. In this implementation, the variance is estimated from the attention history available after the prefill stage. Therefore, the proposed method uses prefill-stage statistics to determine the layer-wise visual KV cache budget before the main compression operation. This design avoids continuously recomputing the global preference score at every decoding step and keeps the additional overhead limited. We emphasize that the variance term measures query-position fluctuation within the encoded multimodal context, rather than online attention evolution during autoregressive decoding.

The hyperparameters τ_1 and τ_2 control the sensitivity of the layer preference score to cross-modal attention entropy and query-position variance, respectively. Unless otherwise specified, we set $\tau_1 = 1.0$ and $\tau_2 = 0.7$ in all experiments. The smoothing constant ϵ is used to avoid zero preference values.

The complete procedure is summarized in Algorithm 1.

Algorithm 1: Proposed Visual KV Cache Allocation and Variance-Aware Token Pruning

Input: Layer-wise attention tensors $\{A^l\}_{l=1}^L$; text token set T ; visual token set V ; visual cache retention ratio ρ ; temperatures τ_1, τ_2 ; smoothing constant ϵ

Output: Compressed visual KV cache

for each layer $l = 1, \dots, L$ **do**

 Obtain the head-aggregated attention representation $\bar{A}^l$;

 Extract A_{TV}^l and A_{VT}^l according to token sets T and V ;

 Compute cross-modal attention entropy E_{CM}^l ;

 Compute cross-modal query-position variance V_{CM}^l from the prefill-stage attention history;

 Compute layer preference $\mathcal{P}_l = (E_{CM}^l + \epsilon)^{\tau_1} (V_{CM}^l + \epsilon)^{\tau_2}$;

Set the total visual cache budget across all layers as $B_{\text{total}} = \rho L |V|$;

Allocate layer-wise visual cache budgets $S_{1:L} = \text{BudgetRouter}(\{\mathcal{P}_l\}_{l=1}^L, B_{\text{total}})$ and adjust them so that $\sum_{l=1}^L S_l = B_{\text{total}}$;

for each layer $l = 1, \dots, L$ **do**

 Compute token importance scores I_n^l for candidate visual tokens using the attention mean–variance criterion;

 Preserve all text KV states and the most recent tokens within the protected recent window;

 Retain the remaining top-scoring non-recent visual tokens according to I_n^l until the layer-wise budget S_l is reached;

 Prune the unselected visual KV states without feature merging;

Return the compressed visual KV cache;

For intra-layer token pruning, the attention mean–variance criterion implemented in the cache eviction module is used. Let $\bar{A}^l$ denote the head-averaged attention matrix in layer l , and let R_q denote the recent query window used for estimating token importance. Text KV states are preserved, and the eviction candidates are restricted to visual key positions that are not protected by the recent-token window. For a candidate visual key token n , its importance score is defined as

$$I_n^l = \frac{1}{|R_q|} \sum_{q \in R_q} \bar{A}_{q,n}^l + \gamma \text{Var}_{q \in R_q}(\bar{A}_{q,n}^l), \quad (1)$$

where the first term measures the average attention received by visual token n , and the second term measures its attention variability across recent query positions. The coefficient γ controls the contribution of the variance term. In our implementation, the recent window size is set to 100 and γ is set to 160.0. The most recent tokens are always preserved, and the remaining non-recent visual tokens are retained according to I_n^l until the layer-wise budget is reached. Unselected visual KV states are directly pruned without feature merging. In this way, text tokens are not selected as eviction candidates.

The budget router normalizes the layer preference scores and converts them into integer layer-wise cache budgets used by the eviction module. Specifically, the total visual budget across all layers is $B_{\text{total}} = \rho L |V|$, and the average per-layer budget is $\rho |V|$. The router assigns larger S_l values to layers with larger preference scores and smaller S_l values to layers with smaller preference scores. After integer rounding, the final budgets are adjusted so that $\sum_{l=1}^L S_l = B_{\text{total}}$, with each layer budget constrained to the valid visual token range.

3.2. Complexity and Memory Analysis

For a Transformer with L layers, hidden dimension d , and visual cache length $|V|$, the visual KV cache storage is proportional to $2L|V|d$ under a fixed numerical precision, where the factor 2 corresponds to keys and values. When the visual cache retention ratio is set to ρ , the proposed method reduces the visual KV cache storage to approximately $2L\rho|V|d$, excluding small metadata overhead. Therefore, the theoretical visual KV cache reduction is approximately $1 - \rho$. When $\rho = 0.2$, only 20% of the visual KV cache is retained, corresponding to an approximate 80% reduction in visual KV cache storage.

The additional computational overhead mainly comes from cross-modal attention entropy estimation, query-position variance estimation, token ranking, and KV index selection. Since the layer preference score is computed from prefill-stage attention statistics and token pruning is applied only when compression is triggered, the overhead is kept limited. In addition to the theoretical cache-storage reduction, measured KV cache GPU memory and decoding latency are reported in the experiments. These measurements provide an initial system-level profile, although broader deployment-oriented evaluation, including prefill latency, throughput, peak end-to-end GPU memory, and batch-size scaling, remains future work.

3.3. Deployment Scenarios and Input Modality Composition

The proposed method is mainly intended for memory-constrained multimodal long-context inference, such as long-video understanding, multi-image reasoning, and cloud-based or edge-assisted MLLM serving where visual tokens dominate the KV cache footprint. In offline batch analysis, the method can reduce visual cache storage and potentially allow larger effective context lengths or batch sizes. In real-time streaming scenarios, the prefill-stage statistics and compression trigger would need to be implemented incrementally, and the latency overhead should be carefully profiled before deployment.

The expected benefit also depends on input modality composition. The proposed method is more suitable for image-dominant or video-dominant inputs, where visual redundancy is substantial and sparse visual evidence may appear across the visual context. For text-heavy inputs, the benefit is expected to be smaller because the method mainly compresses visual KV states while preserving text tokens. For tasks requiring dense global visual perception, aggressive compression may remove useful background information; therefore, a larger visual cache retention ratio may be preferable.

3.4. Variance-Aware Visual KV Cache Allocation

Our core hypothesis is that an effective visual KV cache allocation strategy should consider two complementary signals: cross-modal interaction breadth and query-position attention fluctuation.

3.4.1. Cross-Modal Attention Entropy (E_{CM}): Estimating Interaction Breadth

Following MEDA, which highlights significant variation in cross-modal dependency across model layers, we use cross-modal attention entropy to estimate the distributional breadth of text–vision interaction. Specifically, entropy is computed from the text-to-vision attention matrix A_{TV}^l and the vision-to-text attention matrix A_{VT}^l :

$$E_{CM}^l = \frac{1}{2} \left(\frac{1}{|T|} \sum_{q \in T} H(\bar{A}_{TV,q,:}^l) + \frac{1}{|V|} \sum_{q \in V} H(\bar{A}_{VT,q,:}^l) \right), \quad (2)$$

where H denotes Shannon entropy, and $|T|$ and $|V|$ denote the numbers of text and visual tokens in the current context, respectively.

The normalization by $|T|$ and $|V|$ accounts for modality imbalance. In multimodal inputs, the number of visual tokens often exceeds the number of text tokens by a large margin. Without normalization, the entropy metric could be dominated by the larger modality and may not reflect the average cross-modal interaction pattern in each layer. By averaging the row-wise entropy values within each direction, the metric estimates the distributional breadth of text-to-vision and vision-to-text attention in a comparable manner.

Intuitively, high cross-modal attention entropy indicates that a layer distributes attention over a broader set of cross-modal tokens, suggesting that a larger visual cache budget may be needed to preserve broad visual context. Conversely, low entropy indicates that the layer concentrates on a smaller set of cross-modal anchors, making more aggressive compression possible.

To address the limitations of cross-modal attention entropy alone, the proposed method uses prefill-stage attention statistics to estimate cross-modal query-position fluctuation. As illustrated in Figure 2, MEDA mainly relies on cross-modal attention entropy to measure interaction breadth, whereas the proposed method additionally measures how cross-modal attention varies across query positions in the encoded multimodal context. This allows the model to compute a variance-aware preference score $\mathcal{P}_l$ before visual KV cache allocation.

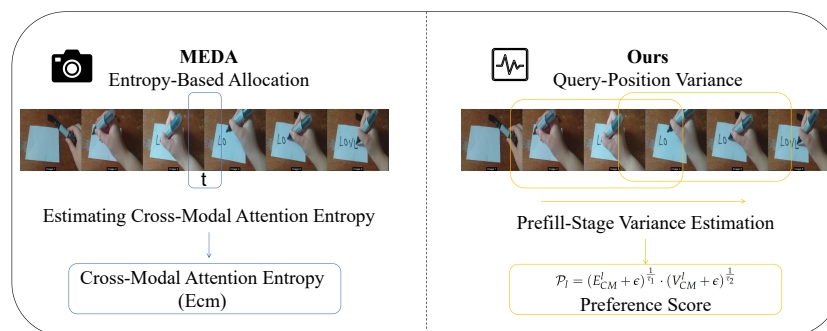


Figure 2. Conceptual comparison between MEDA and the proposed method. MEDA mainly relies on cross-modal attention entropy, whereas the proposed method additionally considers prefill-stage cross-modal query-position variance.

3.4.2. Cross-Modal Query-Position Variance (V_{CM}): Measuring Query-Dependent Attention Variation

Cross-modal attention entropy alone may be insufficient because some visual cues receive weak average attention but remain important for specific reasoning queries. Inspired by variance-based cache analysis in prior work, we introduce a cross-modal query-position variance metric. Unlike language-only variance metrics, this metric is computed from cross-modal attention matrices and is used to estimate how attention to visual evidence varies across query positions in the encoded multimodal context.

The variance metric is defined as

$$V_{CM}^l = \frac{1}{|V|} \sum_{k \in V} \text{Var}_{q \in T}(\bar{A}_{TV,q,k}^l) + \frac{1}{|T|} \sum_{k \in T} \text{Var}_{q \in V}(\bar{A}_{VT,q,k}^l). \quad (3)$$

Here, Var_q denotes variance over query positions in the prefill-stage attention history, rather than variance over autoregressive decoding steps. The index k represents the key position. In the first term, variance is computed over text queries attending to visual keys. In the second term, variance is computed over visual queries attending to text keys. Therefore, V_{CM}^l measures query-position attention variation within the encoded multimodal input, rather than online temporal evolution during decoding.

A higher V_{CM}^l indicates that attention to certain cross-modal keys varies more strongly across query positions. Such visual evidence may not dominate average attention, but it can become important for specific retrieval, comparison, or state-change queries. If allocation were based only on cross-modal attention entropy or mean attention, these sparse query-dependent visual cues could receive insufficient cache budget under low-cache-budget inference.

Figure 3 illustrates this intuition. In layers with relatively low entropy but high query-position variance, some tokens may appear redundant under average attention but become important for specific query positions. The proposed method uses V_{CM} to allocate more cache budget to such layers, increasing the chance that sparse but informative visual cues are retained by the subsequent pruning step. This mechanism may help preserve visual evidence that is easy to overlook under cross-modal attention entropy-based allocation, although the causal importance of such cues still requires further validation.

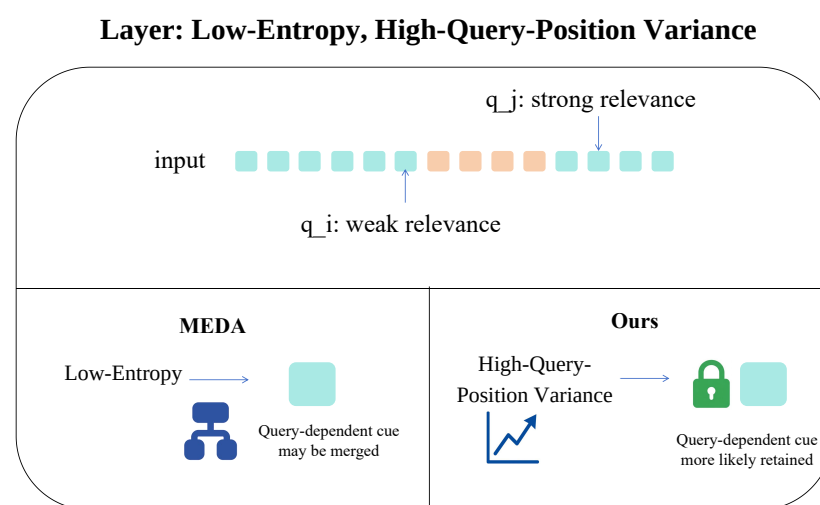


Figure 3. Illustration of a low-entropy, high-query-position-variance layer. Query-dependent visual cues may be under-preserved by entropy-based allocation with token merging, while incorporating query-position variance makes them more likely to be retained during token pruning.

3.4.3. Comprehensive Preference Score ($\mathcal{P}_l$)

Finally, cross-modal interaction breadth, measured by cross-modal attention entropy, and query-position attention variation, measured by cross-modal query-position variance, are combined to construct the layer preference score

$$\mathcal{P}_l = (E_{CM}^l + \epsilon)^{\tau_1} \cdot (V_{CM}^l + \epsilon)^{\tau_2}, \quad (4)$$

where τ_1 and τ_2 control the sensitivity of the layer preference score to cross-modal attention entropy and query-position variance, respectively. Unless otherwise specified, we set $\tau_1 = 1.0$ and $\tau_2 = 0.7$ in all experiments. Larger values increase the contribution of the corresponding signal, while ϵ avoids zero preference values.

3.5. Implementation Pipeline: Global Allocation and Local Token Pruning

In the implementation, the proposed method adopts a conservative “Global Budget Allocation + Local Token Pruning” pipeline. Unlike token-merging-based compression strategies, this design first allocates layer-wise visual KV cache budgets using prefill-stage cross-modal statistics and then performs local pruning within each layer by directly evicting low-scoring visual KV states.

3.5.1. Step 1: Modality-Aware Visual Token Slicing

At the inference entry point, the input structure is first parsed to identify text and visual token intervals. Multimodal image placeholders are used to locate the index boundaries of visual tokens. Unlike methods that treat the entire context as a monolithic sequence, the proposed method explicitly separates text and visual token regions.

This modality-aware slicing operation serves two purposes. First, it provides the basis for computing cross-modal attention matrices, including A_{TV} and A_{VT} . Second, it defines the operational scope of compression. Since text tokens typically have high semantic density and visual tokens often contain substantial spatial redundancy and frame-level redundancy in video inputs, the KV eviction operation is applied mainly to visual token sets. Text KV states are preserved, and only redundant visual KV states are considered as eviction candidates after the protected recent-token window is excluded. This reduces visual KV cache storage while preserving core textual semantics.

3.5.2. Step 2: Global Budget Allocation via Prefill-Stage Statistics

After the prefill stage, the system has access to the attention statistics of the encoded multimodal context. The layer preference score $\mathcal{P}_l$ is then computed for each layer, and the corresponding visual KV cache budget S_l is determined by the budget router

$$B_{\text{total}} = \rho L |V|, \quad S_{1:L} = \text{BudgetRouter}(\{\mathcal{P}_l\}_{l=1}^L, B_{\text{total}}), \quad \sum_{l=1}^L S_l = B_{\text{total}}. \quad (5)$$

where B_{total} denotes the total visual KV cache budget across all Transformer layers, rather than the budget of a single layer. The average per-layer budget is therefore $\rho |V|$, while individual layer budgets S_l are dynamically adjusted according to the layer preference scores. The $\text{BudgetRouter}(\cdot)$ denotes the implementation-specific normalization, rounding, and residual-adjustment procedure that converts continuous layer preferences into integer cache budgets.

This design allows query-position attention variation to participate in budget allocation. It extends MEDA's cross-modal attention entropy-based allocation with a variance-aware metric computed from prefill-stage cross-modal attention. Consequently, for layers that respond to sparse but important visual cues, additional budget can be assigned even when average attention is not dominant.

3.5.3. Step 3: Intra-Layer Variance-Aware Token Pruning

During decoding, once KV cache compression is triggered, the proposed method performs intra-layer visual token pruning according to the allocated layer-wise budget. For each layer, text KV states and recent tokens are always preserved to maintain core textual semantics and local generation continuity. For the remaining non-recent visual tokens, the attention mean-variance importance score I_n^l is computed, and the top-scoring visual tokens are retained until the layer-wise budget is reached.

This design follows a pure KV eviction strategy rather than feature merging. In other words, unselected visual key and value states are directly removed from the KV cache, and no discarded visual token is assigned to or fused with a retained token. This makes the compression operation simple and reproducible while preserving the original representations of retained KV states.

3.6. Potential Robustness Implications and Limitations

Although the proposed method is primarily designed for efficient visual KV cache allocation, reducing redundant visual tokens may have potential implications for robustness. Long multimodal contexts often contain repeated visual patterns, background patches, or weakly relevant visual details. These elements may act as distractors during retrieval or fine-grained reasoning. By assigning larger budgets to layers with stronger cross-modal attention entropy and query-position variance, the proposed method may help retain sparse informative cues while reducing redundant visual context.

However, these potential robustness effects should be interpreted cautiously. This paper does not directly evaluate adversarial attacks, hallucination rates, denial-of-service resistance, or security-monitoring performance. Therefore, we do not claim that the proposed method provides verified adversarial robustness or guaranteed hallucination suppression. Instead, these aspects are viewed as possible downstream implications of redundancy reduction. Dedicated robustness benchmarks, perturbation studies, and long-video security-monitoring evaluations are required before stronger conclusions can be drawn.

This distinction is important in high-stake applications such as security monitoring, medical analysis, or educational assessment. Since the proposed method modifies the retained visual evidence through variance-aware KV eviction, it may also affect post hoc interpretability and evidence tracing. Future work should investigate whether the retained KV states can still support reliable attribution to the original visual tokens and whether the compressed cache preserves sufficient evidence for auditability.

4. Experiments

To provide an initial evaluation of the proposed method, we conduct experiments on the MileBench benchmark [15] using LLaVA-v1.5-7B [16] as the main backbone. All experiments are conducted on a workstation equipped with two NVIDIA RTX A4500 GPUs (NVIDIA Corporation, Santa Clara, CA, USA). Unless otherwise specified, compressed methods are evaluated at a 20% visual KV cache retention ratio, i.e., only 20% of the visual KV cache is retained. This setting is used to examine whether the proposed method can preserve task performance under a low-cache-budget regime. Since the evaluation is still limited to a small number of backbones and the cache budget analysis is performed on representative subtasks rather than the full benchmark, the following results should be interpreted as preliminary evidence for the proposed mechanism rather than exhaustive empirical validation. All reported task scores are deterministic point estimates. Because repeated runs, confidence intervals, and statistical significance tests are not included, cases in which compressed inference numerically exceeds the full-cache reference are reported as point-estimate observations rather than evidence of statistically significant superiority.

The experimental results are shown in Table 2.

Table 2. Performance comparison on MileBench using LLaVA-v1.5-7B. Full cache denotes inference without KV cache compression, while all compressed methods retain 20% of the visual KV cache. T-1–T-4 and S-1–S-5 denote MileBench task groups, NH denotes Needle-in-a-Haystack, and I-1 denotes image retrieval. Bold values indicate the highest point estimate among compressed methods.

Method	T-1	T-2	T-3	T-4	S-1	S-2	S-3	S-4	S-5	NH	I-1
Full Cache	37.00	45.25	30.75	37.88	59.63	32.83	11.30	24.05	62.0	5.00	3.70
H ₂ O	36.50	45.25	28.25	36.50	55.25	30.50	9.50	22.80	60.0	1.67	3.30
SnapKV	36.80	45.25	29.80	36.80	54.60	31.50	10.60	21.90	60.0	1.90	3.50
MEDA	37.20	45.25	30.75	38.38	59.38	32.83	10.03	25.00	63.5	5.31	4.30
Ours	36.83	45.25	31.25	39.50	59.50	32.50	11.70	26.40	62.0	5.94	4.80

When retaining only 20% of the visual KV cache, the proposed method achieves an average score of 32.33, which is close to the full-cache reference score of 31.76 in this deterministic single-run setting. Compared with MEDA, the proposed method obtains higher point estimates on T-3, T-4, S-1, S-3, S-4, NH, and I-1 by 0.50, 1.12, 0.12, 1.67, 1.40, 0.63, and 0.50 points, respectively. Compared with H₂O and SnapKV, the proposed method also achieves higher average point-estimate performance across the evaluated subtasks, with average scores of 32.33 for the proposed method, 31.99 for MEDA, 30.24 for SnapKV, and 29.96 for H₂O. These results suggest that incorporating cross-modal query-position variance may provide useful point-estimate information beyond cross-modal attention entropy-based allocation and attention-heavy-hitter selection. However, because several improvements are modest in absolute value and no repeated-run statistical testing is included, the results are interpreted as preliminary task-level evidence rather than conclusive statistical validation.

4.1. Additional Backbone Evaluation

To partially address the concern that the main experiments use a single backbone, we further evaluate the proposed method on InternVL2.5-8B [17] using representative subtasks. This experiment is intended to provide additional evidence of backbone transferability, rather than a complete multi-backbone evaluation over the full benchmark.

As shown in Table 3, the proposed method remains competitive on InternVL2.5-8B under the same deterministic evaluation protocol. It achieves the highest average point estimate among the compressed methods and remains close to the full-cache reference. Compared with MEDA, the proposed method increases the average point estimate from 36.53 to 41.83, with numerical gains on MovingDirection, MMCoQA, ALFRED, and TextNeedle. These results suggest that the benefit of incorporating query-position variance may not be limited to LLaVA-v1.5-7B, although the small numerical difference over the full-cache reference should not be interpreted as statistically significant superiority. Nevertheless, since this experiment is conducted on representative subtasks rather than the full benchmark, broader multi-backbone evaluation remains future work.

Table 3. Additional evaluation on InternVL2.5-8B. Full cache denotes inference without KV cache compression, while all compressed methods retain 20% of the visual KV cache. The reported subtasks are representative tasks selected from the broader MileBench task groups. Bold values indicate the highest point estimate among compressed methods.

Method	Moving Direction	State Change	CLEVR-Change	MMCoQA	ALFRED	TextNeedle	Avg.
Full Cache	50.5	56.0	24.7	46.0	35.3	37.5	41.67
H ₂ O	48.0	47.5	22.0	36.0	24.8	4.1	30.40
SnapKV	50.0	49.0	17.2	37.5	26.9	14.1	32.45
MEDA	50.0	55.0	21.8	39.0	25.6	27.8	36.53
Ours	52.5	55.5	22.6	48.5	34.4	37.5	41.83

4.2. Cache-Ratio Sensitivity on Representative Subtasks

To partially address the concern that the main comparison is conducted at a single visual KV cache retention ratio, we further evaluate representative subtasks under different visual cache budgets. This analysis focuses on MMCoQA and TextNeedle, which are representative of multi-image question answering and retrieval-oriented reasoning, respectively. The purpose of this experiment is to provide an initial view of the performance–compression trade-off, rather than a complete budget-sensitivity analysis over all MileBench tasks.

As shown in Figure 4, the proposed method maintains competitive point-estimate performance across different visual KV cache budgets on the selected subtasks. On MM-

CoQA, the proposed method remains close to the full-cache reference under moderate cache budgets and performs competitively against the baselines. On TextNeedle, the proposed method shows a clearer numerical advantage under low-cache-budget settings, suggesting that query-position variance may help preserve sparse query-dependent visual cues when retrieval-oriented evidence is easy to discard. These results provide an initial point-estimate indication that the proposed allocation signal is not limited to a single cache retention ratio. Nevertheless, because this analysis is conducted on representative subtasks rather than the full benchmark, a denser evaluation over more cache ratios and tasks remains future work.

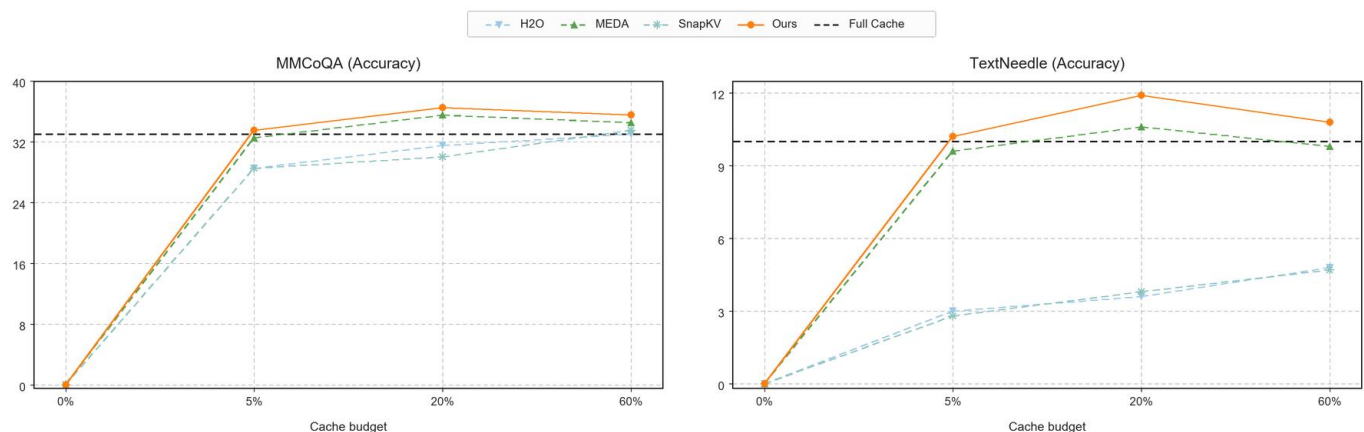


Figure 4. Accuracy comparison under different visual KV cache retention ratios on representative subtasks. The dashed horizontal line denotes the full-cache reference performance.

4.3. Internal Ablation Study of Layer Allocation Signals

To clarify the purpose of this ablation, Table 4 reports an internal study of the proposed pipeline, rather than a direct comparison with MEDA. The T-series and S-series task groups in Table 2 each contain multiple subtasks. We therefore select six representative subtasks from these broader groups to examine how different layer allocation signals affect the proposed method. In particular, the entropy-only variant uses cross-modal attention entropy as the sole signal for layer budget allocation, whereas the variance-only variant uses query-position variance as the sole signal. Both variants still apply the same pruning mechanism after allocation. Therefore, although the entropy-only variant and MEDA both rely on cross-modal attention entropy for allocation, they are not equivalent because the subsequent compression operations are different.

For this ablation, the full entropy–variance allocation uses the default setting $\tau_1 = 1.0$ and $\tau_2 = 0.7$. The entropy-only variant disables the query-position variance signal, equivalent to setting $\tau_2 = 0$ while retaining the entropy signal. The variance-only variant disables the entropy signal, equivalent to setting $\tau_1 = 0$ while retaining the variance signal.

Table 4 analyzes how different layer allocation signals affect the proposed pipeline, and it should be read as an internal component study rather than as evidence that compressed inference generally outperforms full-cache inference. When only cross-modal attention entropy is used for layer allocation, the average score decreases from 24.87 to 19.25. When only query-position variance is used, the average score decreases to 19.53. The gap is particularly noticeable on MMCoQA and TextNeedle, where entropy–variance allocation yields higher point estimates than entropy-only allocation by 12.0 and 10.3 points, respectively. These results suggest that query-position variance can provide complementary point-estimate information to cross-modal attention entropy-based allocation within the proposed pipeline. The comparable drop observed with variance-only allocation further suggests that neither signal alone is sufficient for effective layer budget allocation in this representative-subtask setting; instead, the two signals appear more effective when used jointly.

Table 4. Internal ablation study of layer allocation signals at the 20% visual KV cache retention ratio. The selected subtasks are representative tasks drawn from the broader T-series and S-series task groups reported in Table 2.

Method	Moving Direction	State Change	CLEVR-Change	MMCoQA	ALFRED	TextNeedle	Avg.
Full Cache	31.5	28.0	16.5	33.0	15.1	10.0	22.35
Ours (Ent.-only)	31.5	30.0	15.1	24.5	12.8	1.6	19.25
Ours (Var.-only)	31.5	30.5	14.9	25.0	13.1	2.2	19.53
Ours (Ent.-Var.)	33.0	33.5	18.0	36.5	16.3	11.9	24.87
Ent.-Var. vs. Ent.-only	+1.5	+3.5	+2.9	+12.0	+3.5	+10.3	+5.62
Ent.-Var. vs. Var.-only	+1.5	+3.0	+3.1	+11.5	+3.2	+9.7	+5.33

Ent. denotes cross-modal attention entropy, and Var. denotes query-position variance.

4.4. System-Level Cache Memory and Decoding Latency

In addition to task accuracy, measured KV cache GPU memory and decoding latency are reported in Table 5. At the 20% cache setting used in the main experiments, measured KV cache memory decreases from 1.28 GiB to 0.26 GiB, corresponding to a 79.7% reduction. Decoding latency decreases from 100.28 ms/token to 92.85 ms/token. The latency reduction is more moderate than the memory reduction, which is expected because decoding time also includes computation outside KV cache storage, such as model computation, token generation overhead, and framework-level scheduling. These results show that the proposed compression strategy provides measured memory savings and moderate decoding-latency benefits, rather than only theoretical cache reduction.

Table 5. Measured decoding latency and KV cache GPU memory under different visual KV cache retention ratios.

Cache Setting	Decode Latency (ms/Token)	KV Cache GPU Memory (GiB)	Memory Reduction
Full Cache	100.28	1.28	–
50% Cache	95.19	0.67	47.7%
20% Cache	92.85	0.26	79.7%
5% Cache	88.33	0.06	95.3%

All measurements are obtained on the same workstation equipped with two NVIDIA RTX A4500 GPUs, using the same backbone, precision, batch size, and decoding configuration as the main experiments. Warm-up iterations are used before measurement, and the average decoding latency is reported over the evaluated decoding steps.

4.5. Fine-Grained Visual Relationship Inference

The clearest point-estimate differences appear in the S-3 visual relationship inference task, which includes subtasks such as Spot-the-Diff that require the model to identify subtle differences between similar images. MEDA exhibits a performance drop on this task, decreasing from the full-cache score of 11.30 to 10.03. This result is consistent with the concern that subtle visual differences may not be sufficiently emphasized by cross-modal attention entropy alone and may therefore receive lower retention priority under low-cache-budget inference. In contrast, the proposed method achieves a score of 11.70, which is numerically higher than MEDA and the full-cache reference in this deterministic single-run evaluation. This point-estimate observation suggests that cross-modal query-position variance may help identify subtle visual cues whose average attention is not dominant but whose relevance varies across specific query positions. By assigning higher retention priority to such cues, the proposed method may help preserve fine-grained visual evidence while reducing visual KV cache storage.

4.6. State-Change and Counterfactual Reasoning

In the T-4 counterfactual reasoning and state-change task, the proposed method also achieves the highest point estimate, with a score of 39.50, compared with MEDA (38.38) and the full-cache reference (37.88). Tasks of this type require the model to reason about state transitions and causal relations within visual sequences. Such reasoning may depend on sparse visual evidence that is important for particular query positions. The observed higher point estimate of the proposed method may be associated with its use of query-position attention variation, which allows additional cache budget to be assigned to layers that respond to sparse but important visual cues. The results suggest that removing low-scoring redundant visual KV states may preserve more cache capacity for evidence relevant to state-change reasoning.

4.7. Retrieval-Oriented Point-Estimate Observations Under Compression

In the NH and I-1 tasks, the proposed method achieves scores of 5.94 and 4.80, respectively. Compared with MEDA, the absolute numerical gains are 0.63 and 0.50 points; compared with the full-cache reference, the numerical differences are 0.94 and 1.10 points. These results suggest that, in some deterministic point-estimate cases, retaining all visual tokens may not yield the highest observed score for retrieval-oriented reasoning, possibly because redundant visual context can introduce distractors. By leveraging adaptive budget allocation and variance-aware token pruning, the proposed method may improve the signal-to-noise ratio of the retained visual context under low-cache-budget inference. Nevertheless, the numerical gains are modest in absolute value and should be interpreted as task-dependent evidence rather than a general guarantee of retrieval superiority.

4.8. Neutral and Negative Cases

The proposed method does not improve all subtasks. Slight decreases are observed on T-1, S-2, and S-5 compared with MEDA or the full-cache baseline. These results indicate that variance-aware allocation is not universally beneficial. One possible reason is that some tasks depend on globally distributed visual context rather than sparse query-dependent visual cues. Under such conditions, low-cache-budget inference may remove background information that remains useful for holistic scene understanding. In addition, variance-aware token pruning may remove fine-grained attributes when visually subtle but semantically useful tokens receive low importance scores. Therefore, the proposed method is expected to be more useful for tasks involving sparse evidence, state changes, fine-grained visual differences, and long-range retrieval, while its advantage may be limited for tasks requiring dense global perception.

Because several numerical gains are modest in absolute value, they should not be treated as conclusive evidence of universal superiority. Repeated runs, confidence intervals, statistical significance tests, and full-benchmark evaluations under multiple cache retention ratios would be required to draw stronger empirical conclusions. Accordingly, numerical cases in which compressed inference exceeds the full-cache reference should be regarded as single-run point-estimate observations rather than statistically significant improvements.

5. Limitations

This study has several limitations. First, although the current experiments include an additional evaluation on InternVL2.5-8B, the backbone coverage remains limited. The results therefore do not fully demonstrate whether the proposed method generalizes to a broader range of MLLM architectures. In addition, although we include a cache-ratio sensitivity analysis on representative subtasks, this analysis does not constitute a full budget-sensitivity study over all MileBench tasks. A denser evaluation across more cache

retention ratios, tasks, and backbones remains future work. Second, although measured KV cache GPU memory and decoding latency are reported, the system-level evaluation is still incomplete. A full serving benchmark covering prefill latency, end-to-end peak GPU memory, throughput, concurrent batch-size scaling, and deployment-level scheduling overhead is not yet provided. Third, the current evaluation does not include repeated-run confidence intervals or statistical significance tests. To maintain consistency with existing KV cache compression baselines, the experiments follow a deterministic decoding protocol, and the ablation focuses on controlled component isolation. Some improvements are modest in absolute value and should therefore be interpreted as mechanism-oriented preliminary evidence rather than conclusive statistical validation. Accordingly, numerical cases in which the compressed setting exceeds the full-cache reference should be regarded as single-run point-estimate observations, not as evidence of statistically significant superiority over full-cache inference. Fourth, the discussions of hallucination suppression, adversarial-noise filtering, and security monitoring are potential implications rather than experimentally verified robustness guarantees. Dedicated robustness and deployment evaluations are necessary before the proposed method can be applied in high-stake scenarios. Finally, because the proposed method directly evicts unselected KV states, some fine-grained visual evidence may be removed when the cache retention ratio is very low. Future work should investigate more interpretable cache selection strategies and analyze whether the retained KV states preserve sufficient evidence for post hoc attribution and auditability.

6. Conclusions

This paper studies KV cache optimization for multimodal long-context inference. We propose a cross-modal query-position variance-aware extension of cross-modal attention entropy-based visual KV cache allocation. The key idea is to combine cross-modal attention entropy, which reflects the distributional breadth of text–vision interaction, with prefill-stage query-position variance, which reflects attention variation across query positions in the encoded multimodal context. Based on this preference score, the proposed method allocates visual KV cache budgets across layers and applies variance-aware token pruning to retain high-importance KV states while evicting redundant visual tokens.

Experiments on MileBench using LLaVA-v1.5-7B show that, while retaining only 20% of the visual KV cache, the proposed method achieves point-estimate performance close to the full-cache reference and obtains numerical gains on several tasks related to fine-grained visual relationship inference, counterfactual reasoning, Needle-in-a-Haystack, and image retrieval. Additional evaluations on representative subtasks further show that the proposed method remains competitive under different visual KV cache budgets and on InternVL2.5-8B, providing initial point-estimate evidence beyond a single cache ratio and a single backbone. The internal ablation further shows that entropy–variance allocation yields higher point estimates than both entropy-only and variance-only allocation over six representative subtasks, suggesting that the two signals are complementary within the proposed pipeline. System profiling also shows that, at the 20% cache setting, measured KV cache GPU memory decreases from 1.28 GiB to 0.26 GiB, while decoding latency decreases from 100.28 ms/token to 92.85 ms/token. These results suggest that prefill-stage cross-modal query-position variance may provide a useful complementary signal to cross-modal attention entropy-based allocation, especially when key visual cues are sparse or query-dependent. At the same time, the proposed method does not improve all subtasks, and some improvements are modest in absolute value. Therefore, the current results should be interpreted as mechanism-oriented preliminary evidence rather than a complete demonstration of general superiority, and numerical cases above the full-

cache reference should not be interpreted as statistically significant improvements without repeated-run validation.

Looking ahead, the proposed method may be combined with quantization, hierarchical storage, or hardware-aware cache management to further reduce memory pressure in long-context multimodal systems. Future work should evaluate the method across more MLLM backbones, cache retention ratios, and deployment settings, including measured GPU memory, prefill latency, decoding latency, throughput, and batch-size scaling. In addition, dedicated robustness and interpretability evaluations are needed to determine whether variance-aware cache compression can reliably support high-stake applications such as security monitoring, medical analysis, or educational assessment.

Author Contributions: Conceptualization, S.L. and S.A.; Methodology, S.A.; Software, S.A. and C.Y.; Validation, S.A. and C.Y.; Formal Analysis, S.A. and C.Y.; Investigation, S.A.; Resources, S.L., Y.L. and Y.R.; Data Curation, S.A. and C.Y.; Writing—Original Draft Preparation, S.A.; Writing—Review and Editing, S.L., Y.L., S.A., C.Y. and Y.R.; Visualization, S.A. and C.Y.; Supervision, S.L., Y.L. and Y.R.; Project Administration, S.L. and Y.L.; Funding Acquisition, S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by State Grid Information & Telecommunication Center (Big Data Center) under the project “Key Technology Research on Automated Assessment and Disposal of Data Security Risks Based on Large Language Models in the State Grid Big Data Center”, grant number SGSJ0000HGJS2500036.

Data Availability Statement: Data will be made available on request.

Conflicts of Interest: Shenglong Liu, Yanli Lv and Siyao An are employees of State Grid Information & Telecommunication Center (Big Data Center). The authors declare no other conflicts of interest.

References

1. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*; Curran Associates Inc.: Red Hook, NY, USA, 2017; Volume 30.
2. Li, H.; Li, Y.; Tian, A.; Tang, T.; Xu, Z.; Chen, X.; Hu, N.; Dong, W.; Li, Q.; Chen, L. A survey on large language model acceleration based on kv cache management. *arXiv* **2024**, arXiv:2412.19442.
3. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning (ICML)*; PMLR: Cambridge, MA, USA, 2021; pp. 8748–8763.
4. Liu, Z.; Yuan, J.; Jin, H.; Zhong, S.; Xu, Z.; Braverman, V.; Chen, B.; Hu, X. Kivi: A tuning-free asymmetric 2bit quantization for kv cache. *arXiv* **2024**, arXiv:2402.02750.
5. Kwon, W.; Li, Z.; Zhuang, S.; Sheng, Y.; Zheng, L.; Yu, C.H.; Gonzalez, J.; Zhang, H.; Stoica, I. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*; ACM: New York, NY, USA, 2023; pp. 611–626.
6. Xiao, G.; Tian, Y.; Chen, B.; Han, S.; Lewis, M. Efficient streaming language models with attention sinks. *arXiv* **2023**, arXiv:2309.17453.
7. Zhang, Z.; Sheng, Y.; Zhou, T.; Chen, T.; Zheng, L.; Cai, R.; Song, Z.; Tian, Y.; Ré, C.; Barrett, C.; et al. H₂O: Heavy-hitter oracle for efficient generative inference of large language models. In *Advances in Neural Information Processing Systems*; Curran Associates Inc.: Red Hook, NY, USA, 2023; Volume 36, pp. 34661–34710.
8. Cai, Z.; Zhang, Y.; Gao, B.; Liu, Y.; Li, Y.; Liu, T.; Lu, K.; Xiong, W.; Dong, Y.; Hu, J.; et al. Pyramidkv: Dynamic kv cache compression based on pyramidal information funneling. *arXiv* **2024**, arXiv:2406.02069.
9. Qin, Z.; Cao, Y.; Lin, M.; Hu, W.; Fan, S.; Cheng, K.; Lin, W.; Li, J. Cake: Cascading and adaptive kv cache eviction with layer preferences. *arXiv* **2025**, arXiv:2503.12491.
10. Zhu, J.; Wu, H.; Wang, H.; Li, Y.; Hou, B.; Li, R.; Zhai, J. Fastcache: Optimizing multimodal llm serving through lightweight kv-cache compression framework. *arXiv* **2025**, arXiv:2503.08461.

11. Wan, Z.; Wu, Z.; Liu, C.; Huang, J.; Zhu, Z.; Jin, P.; Wang, L.; Yuan, L. Look-m: Look-once optimization in kv cache for efficient multimodal long-context inference. In *Findings of the Association for Computational Linguistics: EMNLP 2024*; ACL: Stroudsburg, PA, USA, 2024; pp. 4065–4078.
12. Wan, Z.; Shen, H.; Wang, X.; Liu, C.; Mai, Z.; Zhang, M. Meda: Dynamic kv cache allocation for efficient multimodal long-context inference. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*; ACL: Stroudsburg, PA, USA, 2025; pp. 2485–2497.
13. Hang, C.N.; Ho, S.M. Personalized Vocabulary Learning through Images: Harnessing Multimodal Large Language Models for Early Childhood Education. In *Proceedings of the 2025 IEEE Integrated STEM Education Conference (ISEC)*, Princeton, NJ, USA, 15 March 2025; pp. 1–7. [[CrossRef](#)]
14. Mine, Y.; Taji, T.; Okazaki, S.; Takeda, S.; Peng, T.Y.; Shimoe, S.; Kaku, M.; Nikawa, H.; Kakimoto, N.; Murayama, T. Analyzing the performance of multimodal large language models on visually-based questions in the Japanese National Examination for Dental Technicians. *J. Dent. Sci.* **2025**, *20*, 2460–2466. [[CrossRef](#)] [[PubMed](#)]
15. Song, D.; Chen, S.; Chen, G.H.; Yu, F.; Wan, X.; Wang, B. Milebench: Benchmarking mllms in long context. *arXiv* **2024**, arXiv:2404.18532.
16. Liu, H.; Li, C.; Li, Y.; Lee, Y.J. Improved baselines with visual instruction tuning. *arXiv* **2023**, arXiv:2310.03744.
17. Chen, Z.; Wang, W.; Cao, Y.; Liu, Y.; Gao, Z.; Cui, E.; Zhu, J.; Ye, S.; Tian, H.; Liu, Z.; et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv* **2024**, arXiv:2412.05271.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.