DAVID: Dual-Stage Adaptive Vision-Text Integrated Decoupling for Multimodal KV Cache Eviction

Yifeng Gu[1], Jianxiu Jin[1], Kailing Guo[1*], Xiangmin Xu[2, 1],

[1]School of Electronic and Information Engineering, South China University of Technology
[2]Foshan University
eegyf@mail.scut.edu.cn, guokl@scut.edu.cn

Abstract

With the rapid development of multimodal large language models (MLLMs), deploying them on low-resource devices remains challenging. Beyond the model size, long multimodal inputs cause substantial memory overhead in the KV cache, making efficient cache management critical. In this paper, we propose DAVID, a KV cache eviction strategy that adapts to the degree of modality fusion across layers. By analyzing the feature distributions of vision and text tokens, we observe low fusion in early layers and high fusion in deeper layers. Based on this observation, DAVID adopts a decoupled eviction strategy in shallow layers and a super-modal eviction strategy in deeper layers. To support this dynamic switching, we design a lightweight metric that quantifies cross-modal fusion and uses a threshold to determine which layers require decoupling. Experimental results show that DAVID achieves state-of-the-art performance on multiple benchmarks and offers a new perspective on KV cache eviction for MLLMs.

Code — https://github.com/MeL0D/DAVID

Introduction

Recent advances in large language models (LLMs) have rapidly extended from unimodal text-based systems (Achiam et al. 2023; Touvron et al. 2023a,b; Zhang et al. 2022) to multimodal large language models (MLLMs) that jointly process visual and textual inputs (Li et al. 2025; Liu et al. 2023, 2024a; Bai et al. 2025). Vision-language models (VLMs), such as LLaVA (Liu et al. 2023, 2024a), integrate Contrastive Language-Image Pre-training (CLIP)-based vision encoders (Radford et al. 2021) with LLM backbones through a lightweight projector, enabling unified reasoning across modalities. However, as VLM inputs evolve from single low-resolution images and short text to multiple high-resolution images and long text (Bai et al. 2025; Team et al. 2025; Li et al. 2024b), inference costs grow sharply. A major bottleneck arises from the Key–Value (KV) cache, which stores attention states to avoid redundant computation during generation. The KV cache dramatically accelerates inference but consumes substantial GPU memory, making the deployment of VLMs on resource-limited devices challenging.

To address this issue, recent works have explored KV cache compression and eviction. Early studies such as Heavy-Hitter Oracle (H2O) (Zhang et al. 2023) and SnapKV (Li et al. 2024a) introduced token-level importance metrics to discard redundant cache entries while maintaining accuracy. More recent multimodal extensions (Wan et al. 2024, 2025; Pei, Huang, and Xu 2024) explicitly separate vision and text tokens, designing modality-specific eviction strategies. For instance, Look-Once Optimization in KV Cache for efficient Multimodal long-context inference (LOOK-M) (Wan et al. 2024) observed that text KVs tend to dominate attention and proposed merging visual KVs into textual ones based on similarity, while Multimodal Attention Entropy-Guided Dynamic KV Cache Allocation (MEDA) (Wan et al. 2025) further analyzed layer-wise attention distribution to dynamic allocate KV cache budgets. These KV cache eviction approaches for MLLMs focus on disentangling information from different modalities and applying modality-specific retention strategies. However, we empirically find that multimodal KV eviction sometimes performs worse than unimodal methods such as H2O and SnapKV. This counterintuitive finding raises an open question: *Why do modality-separated KV cache eviction strategies, which are explicitly designed for multimodal models, sometimes underperform strategies developed for unimodal LLMs?*

We posit that this limitation stems from a misinterpretation of modality fusion in MLLMs. Prior approaches rigidly decouple vision and text tokens across all layers, assuming a persistent separation between modalities. However, MLLMs concatenate vision and text embeddings at the input layer, and as data propagate through transformer layers, cross-modal attention progressively blends representations. In early layers, visual and textual tokens remain semantically distinct, but in later layers, they form a fused representation space that we term a super-modality, where tokens encode jointly integrated semantics rather than single-modality cues. In such fused stages, forcibly maintaining modality separation disrupts the coherence of contextual reasoning, making the separation of visual and textual information potentially ineffective or even detrimental to generation quality. Conversely, uniform eviction across already fused representations better aligns with the model's information struc-

*Corresponding author

ture.

Building upon this insight, we investigate how the fusion process affects multimodal KV cache eviction and propose a **D**ual-stage **A**daptive **V**ision-text **I**ntegrated **D**ecoupling (DAVID) for multimodal KV cache eviction. DAVID dynamically adapts its retention behavior to the model's fusion depth. In early layers with low cross-modal fusion, it performs modality-decoupled eviction to preserve modality-specific information; in later layers where fusion makes super-modality form, it switches to uniform eviction without explicit modality separation. To enable this transition, we introduce a cross-modal fusion intensity score that quantitatively measures the degree of representation fusion, thereby identifying the optimal layer boundary for switching eviction modes. Our contributions are listed as follows:

- To the best of our knowledge, our work is the first KV-eviction framework that explicitly characterizes layer-wise fusion progression and uses it to guide a dual-stage modality-aware eviction strategy.
- We propose a dynamic KV cache eviction framework that adapts to the layer-wise fusion state of multimodal representations, bridging unimodal and multimodal pruning paradigms.
- We introduce a cross-modal fusion intensity score that quantifies the integration level of visual and textual features, providing a principled signal for switching eviction modes.

Related Work

Efficient Multimodal Large Language Models

Efficiency has become a central objective for MLLMs as they move toward real-world deployment. Typical MLLMs (Liu et al. 2023, 2024a; Li et al. 2025) connect a visual encoder to an LLM through a lightweight projector, making them compatible with efficiency techniques originally designed for LLMs—such as quantization (Frantar et al. 2023; Dettmers et al. 2022), pruning (Ma, Fang, and Wang 2023; Sun et al. 2024), and low-rank approximation (Wang et al. 2025a,b). A key constraint of MLLMs is their longer multimodal input sequence. To address this, prior works have explored reducing visual or textual inputs (Chu et al. 2024; Cao et al. 2024; Huang et al. 2025; Wan et al. 2024) and lowering KV-cache memory costs. These two directions are orthogonal. Our work belongs to the latter: optimizing KV-cache memory during inference without retraining.

Token Compression in Multimodal Large Language Models

Since most multimodal inputs originate from vision encoders, many studies primarily target the reduction of visual tokens (Chu et al. 2024; Shang et al. 2024; Chen et al. 2024a). MobileVLM (Chu et al. 2024) designs an efficient projector that compresses vision tokens during the alignment of image features to text features. LLaVA-Prumerge (Shang et al. 2024) introduces a plug-and-play pruning module between the visual encoder and the projector that adaptively removes unimportant vision tokens. FastV (Chen et al. 2024a) observes that vision token sparsity varies across layers, and thus adopts a strategy of retaining tokens in early layers while pruning them at specific later layers. Dynamic-LLaVA (Huang et al. 2025) argues that compressing text tokens is equally important in MLLMs, and introduces a token importance predictor to jointly reduce both text and vision tokens. These methods effectively reduce sequence length and consequently KV-cache size, though many of them rely on extra modules, architectural modifications, or task-specific retraining, which may limit deployment simplicity. In contrast, our approach identifies KV importance without extra training, enabling seamless real-time integration.

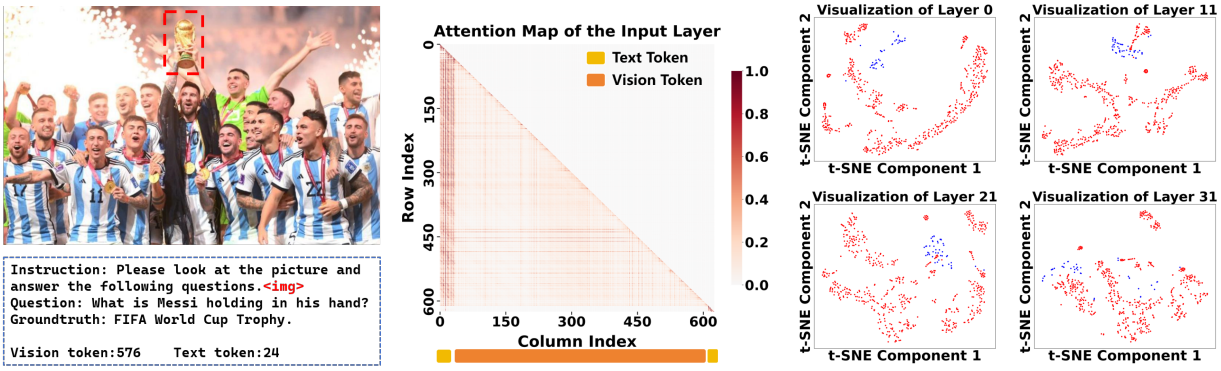
KV Cache Eviction

KV-cache eviction has been extensively studied for unimodal LLMs. H2O (Zhang et al. 2023), uses accumulated attention scores as eviction scores to identify and retain important key-value (KV) pairs. SnapKV (Li et al. 2024a) builds on this by computing accumulated attention scores within an observation window to provide a more accurate assessment of KV importance. NACL (Chen et al. 2024b) and Keyformer (Adnan et al. 2024) further refine KV selection with learned routing or structured sparsity. Multimodal KV eviction introduces new challenges due to modality heterogeneity. LOOK-M (Wan et al. 2024) considers vision tokens to be less important than text tokens. It selects key text tokens based on eviction scores and merges similar vision tokens into them. MEDA (Wan et al. 2025) builds on this by allocating KV budgets across layers according to their importance. CSP (Pei, Huang, and Xu 2024) distinguishes multimodal attention into self and cross attention, calculating their cumulative attentions separately and combining them with weighted sums to derive the multimodal eviction score. While these methods incorporate multimodal cues into the scoring mechanism, they do not explicitly model how fusion evolves across layers.

Motivation

MLLM inputs are longer than those of text-only LLMs, containing rich information from multiple modalities. Research on KV cache eviction for MLLMs (Wan et al. 2024, 2025; Pei, Huang, and Xu 2024) shows that vision and text tokens are heterogeneous, making text-only eviction methods (Zhang et al. 2023; Li et al. 2024a) less suitable for multimodal settings. However, do vision and text tokens truly exhibit heterogeneity? And if so, is this heterogeneity consistent across all layers? Motivated by these questions, we conducted a detailed analysis.

Why is modality decoupling necessary in KV cache eviction for MLLMs? In multimodal tasks, vision and text tokens have distinct sparsity. Fig. 1(a) shows a typical multimodal input, where vision tokens vastly outnumber text tokens. We extracted the first-layer attention scores from LLaVA-1.5-7B with multimodal input, as shown in Fig. 1(b). It shows that text tokens receive consistently higher attention, further highlighting this disparity. Prior unimodal methods (Zhang et al. 2023; Li et al. 2024a) apply uniform



(a) Typical multimodal input. (b) Attention score of the MLLM. (c) Feature clustering in different layers.

Figure 1: (a) illustrates a typical multimodal input, where the text is concise yet information-rich, while the visual input contains substantial redundancy. (b) presents the attention scores at the MLLM input layer, showing that text tokens receive higher attention. (c) depicts the degree of modality fusion across layers, with deeper layers exhibiting more thorough integration of the two modalities.

sparsity rates, risking over-retention or loss of modality-specific information. Therefore, it is necessary to adopt a decoupled KV cache eviction for vision and text modalities.

Rethinking the Impact of Fusion Processes on Tokens in MLLMs. The above analysis highlights the need for modality decoupling in KV eviction. However, decoupling at every layer is unnecessary. Prior methods (Wan et al. 2024, 2025) design eviction strategies based on modality differences, but as fusion progresses, these differences fade. As shown in Fig. 1(c), t-SNE (Van der Maaten and Hinton 2008) visualizations reveal that text and vision tokens, initially separate, gradually merge into shared clusters across layers. This reflects the formation of super-modal tokens, where modality-specific decoupling becomes redundant. Therefore, decoupling is essential in early layers, while unified eviction is more appropriate in deeper layers.

Methodology

In this section, we propose a modality-decoupled KV cache eviction for MLLMs to better identify crucial KV pairs. To account for modality fusion, we introduce a normalized cross attention rate that determines when to transition to unified eviction adaptively. These components form DAVID: a KV cache eviction framework designed for MLLMs.

Decoupled Multimodal KV Cache Eviction

In MLLMs, the typical input consists of concatenated vision and text tokens, while the output is composed of text tokens. These tokens are encoded as Query(Q), Key(K), and Value(V) in the Transformer (Vaswani et al. 2017). Let $Q, K \in \mathbb{R}^{n \times d}$, where n is the total number of tokens and d is the hidden dimension, and $A_{i,j}$ denotes the attention score between the i^{th} query and the j^{th} key, computed as follows.

$$A_{i,j} = Softmax(Q_i K_j / \sqrt{d}) \tag{1}$$

Base on SnapKV (Li et al. 2024a), we use the accumulated attention scores of the most recent r tokens as eviction scores because recent tokens are more relevant to next token. The computation of the eviction scores is as follows.

$$S_i = \sum_{j=n-r}^{n} A_{i,j} \tag{2}$$

Let us denote the vector of S_i by S. Following SnapKV's observation (Li et al. 2024a) that overly dispersed tokens impair generation, we apply maxpool to smooth the eviction scores while preserving the original vector dimensions, thereby maintaining textual continuity. Previous text-only methods (Zhang et al. 2023; Li et al. 2024a) typically set a unified budget B_s, retaining the top B_s KVs with the highest eviction scores as shown in the following.

$$KV_{selected} = TopK(MaxPool(S), B_s) \tag{3}$$

Unified eviction works well for text-only tasks but ignores modality differences in multimodal inputs. Thus, multimodal methods account for varying information across modalities. Previous approaches (Wan et al. 2024, 2025) have found that text tokens are generally more influential than vision ones. This trend was also observed in our analysis. Text tokens typically contribute more to the overall attention distribution and exhibit lower sparsity compared to vision tokens. As a result, these methods prioritize text during generation and merge all vision tokens into the text ones.

However, this assumption doesn't hold universally—fixed priorities may result in significant loss of visual information. Instead, preserving each modality's information only requires decoupling them, rather than enforcing a fixed hierarchy. Let X denote the set of all tokens, $M \in \{V, T\}$ denote different modal, where X^M denotes the set of unimodal tokens, and X_i denotes the i^{th} token. We calculated two eviction scores for vision and text as follows.

$$S_i^M = \sum_{j=n-r}^{n} A_{i,j}, if\ X_i \in X^M\ else\ 0 \tag{4}$$

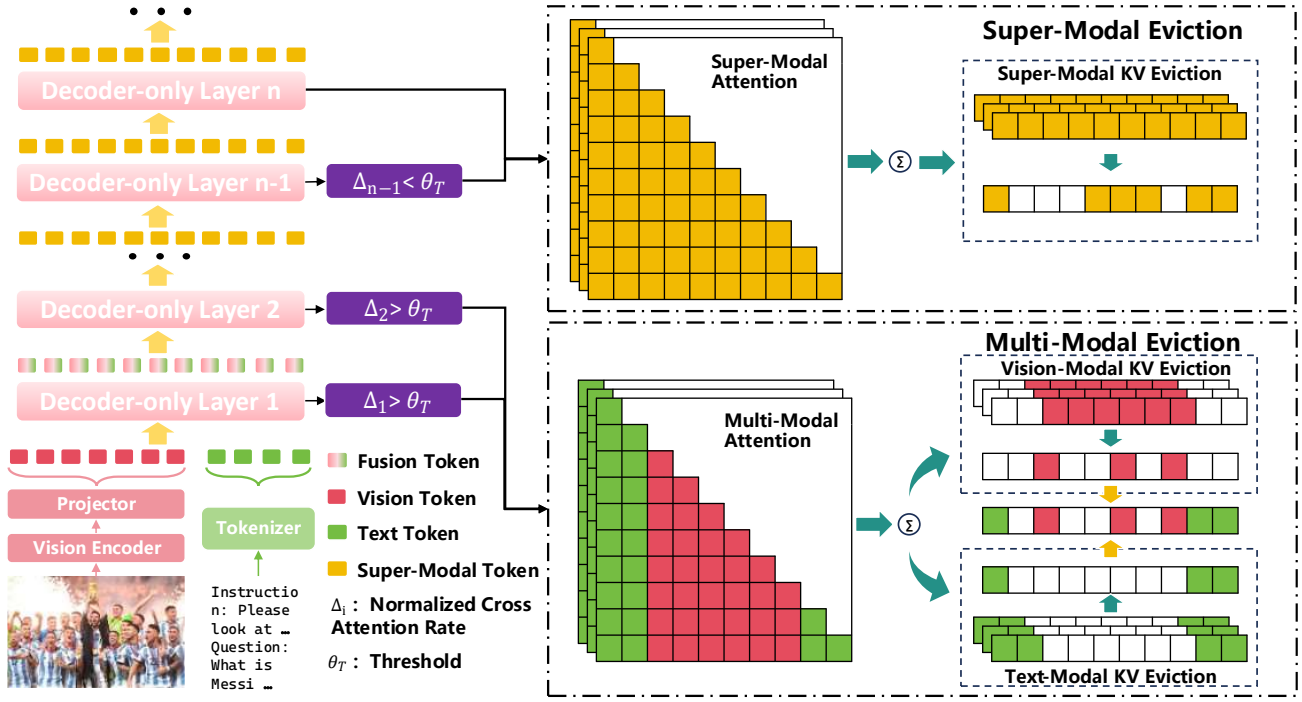


Figure 2: DAVID's algorithm employs two KV eviction strategies: (1) Multi-modal eviction, where modalities are decoupled and scored separately to retain important KVs; and (2) Super-modal eviction, applied after vision and text tokens are fully fused, enabling unified eviction. The switch between them is controlled by the θ parameter—when θ falls below a threshold, Super-modal eviction is triggered.

In practice, we apply $MASK(S, I)$ to zero out scores at positions I, yielding the modality-specific eviction scores. Due to differing sparsity rates, text and vision tokens require separate eviction processes. Based on the decoupled eviction scores, we retain the crucial KVs as follows.

$$KV^{M}_{selected} = TopK(S^{M}, B^{M}_{s}) \tag{5}$$

This indicates that the vision and text modalities retain different numbers of KVs. Under a fixed budget, the ratio $\rho = \frac{B^{V}_{s}}{B^{T}_{s}}$ should be set according to the ratio of vision and text tokens. However, this ratio should not be too large or too small, since this would result in a significant amount of information being lost from one modality. We concatenate the KV pairs from both modalities to obtain the final set of KVs to be retained.

Normalized Cross Attention Rate

The decoupled eviction strategy for handling modality differences was described earlier. As mentioned in the motivation section, the character of fusion between vision tokens and text tokens varies across different layers. As the fusion process progresses, vision tokens and text tokens gradually fuse into a super-modal token. These super-modal tokens can effectively be treated as unimodal, allowing the use of text-only methods for unified eviction. The unimodal eviction strategy is shown in Eq. (3). We need to determine the boundary for switching between eviction strategies.

We define a parameter θ to quantify fusion strength and identify the layers that require modality decoupling. Our goal is to efficiently identify the boundary between decoupled and unified eviction based on θ. In Eq. 1, A_{ij} quantifies the coupling between Q_i and K_j; if they belong to different modalities, it reflects cross-modal coupling. In the shallow layers of VLMs, modalities remain largely separate, making cross-modal coupling essential and leading to high cross-modal attention scores $A_{i,j}$. Previous studies (Chen et al. 2024a; Wan et al. 2025) have shown that the importance of vision tokens decreases with layer depth. This enables us to identify mutations in the average coupling of text and vision, helping us to determine the boundary between decoupling and unified KV cache eviction. Therefore, we propose the normalized cross attention rate (NCAR) as follows:

$$\theta = \frac{n}{n^{V} r} \sum_{i=n-r}^{n} \sum_{j \in V} A_{i,j} \tag{6}$$

Here, we normalize the average cross-modal attention score by $1/n$, where n is the token count. To identify abrupt drops in NCAR and determine when to stop decoupled KV cache eviction, we set a threshold θ_T. We determine the phase transition using the difference in NCAR as follow:

$$\Delta_i = \theta_{i-1} - \theta_i \tag{7}$$

When $\Delta_i < \theta_T$, we stop decoupled eviction and switch to a unified eviction strategy. The DAVID algorithm is presented in Algorithm 1.

Algorithm 1: DAVID Algorithm

Input: Total KV cache budget $B = B_r + B_s$, recent cache budget B_r, selected cache budget B_s, budget ratio ρ, fusion threshold θ_T.
Output: Retained KV indices for each layer and head.
function EVICTION($Prompt$)
    $Flag \leftarrow True$
    $\theta_0 \leftarrow 1$
    for Layer-m in LLMs **do**
        $Q_m, K_m, V_m \in \mathbb{R}^{h \times n \times d}$
        $I_V \leftarrow$ Image Position
        $I_T \leftarrow$ Text Position
        ▷ * signifies an operation shared across the same dim.
        $A_{m,*} \leftarrow Softmax(\frac{Q_{m,*} \times K_{m,*}^T}{\sqrt{d}})$ ▷ $A_m \in \mathbb{R}^{h \times n \times n}$
        $KV_{m,*}^r \leftarrow$ Recent B_r Token
        $S_{m,*} \leftarrow \sum_{Token \in B_r} A_{m,*}$ ▷ $S_m \in \mathbb{R}^{h,n}$
        $\hat{S}_{m,*} \leftarrow MaxPool(S_{m,*})$
        if $Flag$ **then**
            $\theta_{m,*} \leftarrow \frac{n}{n^V} \sum_{i=n-r}^{n} \sum_{j \in V} A_{m,*,i,j}$
            $\theta_m \leftarrow \frac{\sum_k^h \theta_{m,k}}{h}$
            $\Delta_m = \theta_{m-1} - \theta_m$
            if $\Delta_m < \theta_T$ **then**
                $Flag \leftarrow False$
            end if
        end if
        if $Flag$ is $False$ **then**
            $KV_{m,*}^s \leftarrow TopK(\hat{S}_{m,*}, B_s)$
        else
            $\hat{S}^T \leftarrow MASK(\hat{S}, I_V)$
            $\hat{S}^V \leftarrow MASK(\hat{S}, I_T)$
            $B_s^V \leftarrow FLOOR(\frac{B_s}{1+\rho})$ ▷ Round Down
            $B_s^T \leftarrow B_s - B_s^V$
            $KV_{m,*}^{s,V} \leftarrow TopK(\hat{S}^V, B_s^V)$
            $KV_{m,*}^{s,T} \leftarrow TopK(\hat{S}^T, B_s^T)$
            $KV_{m,*}^s \leftarrow [KV_{m,*}^{s,V}, KV_{m,*}^{s,T}]$
        end if
        $KV_{m,i} \leftarrow [KV_{m,i}^s, KV_{m,i}^r]$
    end for
end function

Experiments

In this section, we conduct extensive experiments to validate the effectiveness and efficiency of DAVID. We adopt the LLaVA series (Liu et al. 2023; Li et al. 2025) and Qwen-VL (Bai et al. 2023) as our primary models, given its widespread use and representative architecture. We first compare DAVID's generation performance with other baseline methods. Next, we evaluate its inference speed and analyze key design components through ablation studies. Finally, we provide in-depth discussions and insights into the DAVID framework.

Setup and Baseline

Baseline To evaluate our method, we compare it with two text-based approaches, H2O (Zhang et al. 2023) and SnapKV (Li et al. 2024a), which can be adapted to MLLM KV eviction. We also incorporate multimodal methods, such as LOOK-M (Wan et al. 2024), MEDA (Wan et al. 2025) and CSP (Pei, Huang, and Xu 2024), which reduce KV cache usage by eliminating redundant tokens, and which have demonstrated robust performance in multimodal tasks.

Datasets Previous studies often use MileBench (Dingjie et al. 2024) for evaluation, but its multiple-choice format with single-token answers limits its suitability for assessing eviction performance. To enable a more realistic evaluation, we select open-ended multimodal QA tasks across three datasets, OCRBench (Liu et al. 2024b), MMVet (Yu et al. 2024), and MathVista (Lu et al. 2024), covering OCR, general VQA, and mathematical reasoning. We evaluate these datasets using the widely adopted VLMEvalKit framework (Duan et al. 2024). OCRBench is a comprehensive benchmark for evaluating MLLMs on text recognition, scene-text VQA, document VQA, key information extraction (KIE), and handwritten math expression recognition (HMER). MM-Vet is a widely used benchmark for MLLMs, covering recognition (REC), OCR, knowledge (KNOW), language generation (GEN), spatial reasoning (SPAT), and math, to assess overall model performance. MathVista is a benchmark combining diverse mathematical and visual reasoning tasks, including FQA (Chart QA), GPS (Geometry), MWP (Math Word Problems), TQA (Textbook QA), and VQA. It covers reasoning types such as ALG, ARI, GEO, LOG, NUM, SCI, and STA. OCRBench is measured by recognition accuracy (Acc), while MMVet and MathVista use GPT-4 (Achiam et al. 2023) to assess semantic alignment between predicted and reference answers. To reduce randomness, we average five runs for final results.

Implementation Details All experiments were conducted on a single NVIDIA A800 80GB GPU and NVIDIA GeForce RTX 3090. We set the sparsity rate to 0.2 for all evaluations. The hyperparameter ρ is set to 2 for OCRBench and MMVet, and 1 for MathVista, based on the average vision-to-text token ratio. The fusion threshold θ_T is fixed at 0.3.

Main Results

OCRBench Results We compared DAVID with other methods on the OCRBench (Liu et al. 2024b) dataset. DAVID consistently outperformed all baselines across subtasks and achieved the highest overall score (305 for 7B, 324 for 13B), closely approaching full-cache performance. It also maintained strong results across models of varying scales. We observed that CSP and MEDA, both based on LOOK-M, performed poorly on OCRBench. This is likely due to their overemphasis on the text modality, leading to insufficient attention to visual information—crucial for image-dominated tasks. In contrast, DAVID demonstrates stronger adaptability and generalizability across modalities.

MMVet Results DAVID demonstrates strong performance on the MM-Vet benchmark, outperforming baselines across multiple tasks and model sizes. As shown in Table 2, both LLaVA-1.5-7B and 13B achieve the highest scores in key areas such as REC (44.22 for 7B, 40.21 for 13B), KNOW (20.37 for 7B, 21.67 for 13B), and GEN (19.63 for 7B, 18.75 for 13B), surpassing even the full-cache genera-

Method	Full	H2O	SnapKV	LOOK-M	CSP	MEDA	DAVID	Full	H2O	SnapKV	LOOK-M	CSP	MEDA	DAVID
	LLaVA-1.5-7B							LLaVA-1.5-13B						
Text Recog.	157	154	149	82	127	68	**156**	164	160	159	77	147	81	**160**
Scene VQA	124	102	120	77	101	70	**121**	128	126	128	71	112	79	**128**
Doc VQA	23	13	22	20	23	16	**24**	25	24	28	14	26	16	**29**
KIE	4	2	4	3	4	2	**4**	8	5	7	1	4	3	**7**
HMER	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Final Score	308	271	295	158	255	156	**305**	325	315	322	163	289	179	**324**

Table 1: Results in OCRBench

Method	Full	H2O	SnapKV	LOOK-M	CSP	MEDA	DAVID	Full	H2O	SnapKV	LOOK-M	CSP	MEDA	DAVID
	LLaVA-1.5-7B							LLaVA-1.5-13B						
REC	43.26	38.87	42.35	33.95	41.01	37.43	**44.22**	39.83	32.94	38.55	35.18	37.86	37.86	**40.21**
OCR	20.55	**17.4**	13.24	7.12	15.37	13.7	16.2	29.62	7.59	24.9	12.13	19.35	18.79	**25.09**
KNOW	20.35	11.42	17.73	9.04	17.97	16.67	**20.37**	18.21	2.26	20.00	10.47	15.59	14.88	**21.67**
GEN	19.50	9.00	15.37	4.87	14.00	11.25	**19.63**	17.00	1.12	17.75	10.00	14.37	13.25	**18.75**
SPAT	27.33	19.06	23.73	16.67	23.06	21.33	**25.2**	31.46	16.13	**31.86**	22.93	29.33	28.66	30.53
MATH	15.38	3.84	**8.07**	3.85	3.84	11.15	7.69	18.84	3.84	11.53	11.15	11.53	11.53	**11.53**
Overall Score	31.05	25.64	28.21	20.41	27.61	26.33	**30.18**	32.20	18.76	30.41	23.55	28.11	27.15	**31.28**

Table 2: Results in MMVet

tion accuracy. This suggests that selectively removing tokens irrelevant to generation can enhance overall accuracy. In summary, DAVID achieves the highest overall scores (30.18 for 7B, 31.28 for 13B), demonstrating robust, balanced performance across modalities.

MathVista Results Table 3 highlights DAVID's strong performance on the MathVista benchmark with both LLaVA-1.5-7B and LLaVA-1.5-13B.

For the 7B model, DAVID achieves the highest scores in critical tasks such as ARI (22.66), VQA (34.07), GEO (27.19), ALG (29.18) and LOG (24.32). In terms of overall scores, our approach yielded the best results (26.2). For the 13B model, DAVID further excels in TQA (39.87), ALG (26.33), and STA (24.58), while achieving the highest overall score (27.5) across all methods. DAVID achieved the highest overall performance across both models.

DAVID consistently outperforms other methods, excelling in algebra, numerical, and logical reasoning, while maintaining strong results across diverse multimodal tasks.

Ablation Results

We conducted ablation studies on OCRBench to analyze the effectiveness of each component in DAVID. Our design consists of two key components: the Decoupling Eviction (DE) module and the Normalized Cross Attention Rate (NCAR) judgment. These components operate sequentially, with NCAR relying on DE to take effect. Accordingly, our ablation experiments evaluate DAVID's performance under three settings: without either component, with only DE enabled, and with both DE and NCAR. As shown in the table 4, without either component, the final score is 295. Introducing DE alone improves the score by 3, indicating that applying decoupled eviction across all layers still provides some benefit. With NCAR added, the score rises to 305, 7 points higher than using DE alone. This supports our view that while modality decoupling is beneficial, identifying which layers require decoupling is crucial for optimal performance.

Impact Across Different Budget

To evaluate DAVID's effectiveness under varying budget conditions, we compared its performance at MM-Vet with other methods across different sparsity rates. As shown in Fig. 3, all methods exhibit a performance decline as the budget decreases, which is expected. However, DAVID maintains state-of-the-art performance under almost all budgets. This clearly highlights the superiority of our method. Notably, at a 60% sparsity rate, DAVID even outperforms the full-cache baseline—likely because removing redundant vision tokens helps the model focus more effectively on critical information.

Efficiency Analysis

Table 5 shows the efficiency of DAVID. We took the first 20 samples from the MathVista dataset and analysed their speed and memory usage on an NVIDIA GeForce RTX 3090.

As shown in Table 5, compared to the full-cache model, DAVID significantly reduces GPU memory usage and improves inference speed by eliminating redundant KV pairs, thereby lowering the computational burden. Moreover, our earlier performance results demonstrate that DAVID maintains text generation quality comparable to the full-cache baseline. These results highlight the practical effectiveness of our method.

Compatibility with Text-only Eviction Experiment

As highlighted earlier, our method is fully compatible with text-only approaches. In this section, we conduct extended experiments to demonstrate how our method can be integrated with text-only KV cache eviction strategies.

Our eviction score can be adapted to the eviction scores of other text-only methods, ensuring compatibility. As shown

Method	Full	H2O	SnapKV	LOOK-M	CSP	MEDA	DAVID	Full	H2O	SnapKV	LOOK-M	CSP	MEDA	DAVID
	LLaVA-1.5-7B							LLaVA-1.5-13B						
SCI	33.6	31.96	34.42	**36.88**	33.6	33.6	33.6	37.7	**38.52**	36.06	37.7	36.8	37.7	36.7
TQA	36.7	36.71	36.71	**39.87**	36.07	36.7	36.7	39.87	38.6	39.24	39.24	39.24	39.8	**39.87**
NUM	20.13	18.05	19.44	**20.83**	20.13	19.44	20.14	27.08	24.3	26.38	25.69	27.1	27.7	**27.78**
ARI	22.38	20.11	20.09	20.96	22.37	22.09	**22.66**	25.79	23.2	25.77	23.23	26.34	25.77	**26.62**
VQA	34.08	31.28	33.51	32.4	34.07	33.51	**34.07**	36.87	35.75	36.31	35.75	36.87	37.43	**37.43**
GEO	21.34	20.5	21.33	22.17	22.17	22.17	**27.19**	21.75	21.75	21.33	23.01	24.68	20.08	**24.68**
ALG	25.27	25.62	24.91	25.97	25.62	25.97	**29.18**	23.84	24.12	23.48	25.27	26.33	22.77	**26.33**
GPS	22.6	22.11	22.59	23.07	23.55	23.55	**28.84**	21.15	21.63	20.67	23.07	24.51	19.23	**24.51**
MWP	12.9	11.82	12.9	12.9	12.9	12.9	**12.9**	16.13	14.51	16.67	13.97	**17.7**	16.12	17.20
LOG	21.62	21.62	21.6	21.62	21.62	21.62	**24.32**	13.51	13.51	16.21	16.21	**18.91**	16.21	16.21
FQA	22.67	23.42	21.93	**24.9**	22.67	22.67	22.67	23.42	22.67	23.42	22.3	23.04	22.3	**23.42**
STA	20.93	**20.93**	20.26	22.25	20.93	20.93	20.60	24.25	22.25	24.25	21.26	24.25	22.92	**24.58**
Overall Score	25.1	24.5	24.8	26	25.2	25.2	**26.2**	26.6	25.8	26.4	26.00	27.4	26.00	**27.5**

Table 3: Results in MathVista

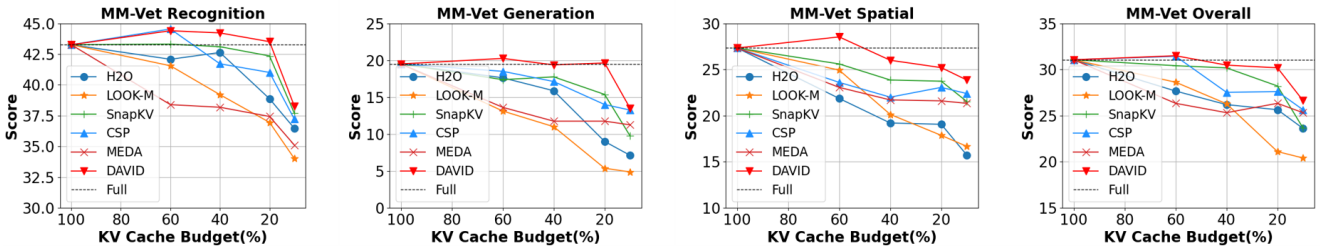


Figure 3: Comparison of eviction performance under different budgets.

	TREC	SVQA	DVQA	KIE	HMER	Final
w/o DE&NCAR	149	120	22	4	0	295
w/o NCAR	149	121	23	5	0	298
DAVID	156	121	24	4	0	305

Table 4: Ablation Experiments

Method	Budget	Decoding Latency	GPU Memory
Full Cache	100%	26.23ms/token	1.26GB
DAVID	60%	22.82ms/token	0.77GB
DAVID	20%	19.17ms/token	0.34GB
DAVID	5%	16.46ms/token	0.18GB

Table 5: Efficiency Analysis

in Table 6, experiments using LLaVA-1.5-7B on OCRBench indicate that using DAVID based on H2O improves our final score by 14 points. Our framework enhances text-only eviction by incorporating modality decoupling, addressing the impact of modality fusion on eviction. Once fusion reaches a certain level, we adopt a unified eviction. It allows our framework to extend text-only methods to multimodal settings, showcasing strong generalization capabilities.

	TREC	SVQA	DVQA	KIE	HMER	Final
H2O	154	102	13	2	0	271
H2O(DAVID)	151	111	19	4	0	285

Table 6: Compatibility with Text-only Eviction Experiment

	Full	SnapKV	DAVID
OCRBench	486	469	478
MMVet	42.15	38.43	40.31

Table 7: Results in Qwen-VL

Extended Experiments on Other MLLM Models

DAVID can be effectively extended to other MLLMs that concat visual and textual tokens. Qwen-VL (Bai et al. 2023) serves as an alternative implementation of this architecture, differing mainly in its visual encoder design. We evaluate DAVID on Qwen-VL at 0.4 sparsity rate, where it achieves a 9 points improvement over SnapKV on OCRBench and a 1.88 points gain on MMVet. These results demonstrate that our method can be readily applied to models with similar architectures.

Conclusion

In this paper, we propose **DAVID**, a KV cache eviction method tailored for MLLMs to reduce memory consumption during generation. Unlike previous unimodal and multimodal approaches that overlook the modality fusion process and apply uniform eviction strategies across all layers, DAVID dynamically adapts to the degree of fusion at each layer. By distinguishing between decoupled modal eviction and super-modal eviction, DAVID more effectively retains important tokens. Experimental results show that DAVID achieves SOTA performance across multiple benchmarks.

Acknowledgments

This work is supported in part by the National Key R&D Program of China under Grant 2022YFB4500600, in part by the the Science and Technology Projects of Guangzhou under Grant 2023B03J1300, in part by the Taihu Lake Innovation Fund for the School of Future Technology of South China University of Technology, under Grant 2024B105611002, in part by the National Natural Science Foundation of China under Grant 61802131, and in part by the Guangdong Provincial Key Laboratory of Human Digital Twin, under Grant 2022B1212010004.

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

Adnan, M.; Arunkumar, A.; Jain, G.; Nair, P. J.; Soloveychik, I.; and Kamath, P. 2024. KeyFormer: KV cache reduction through key tokens selection for efficient generative inference. *In Proceedings of Machine Learning and Systems*, 6: 114–127.

Bai, J.; Bai, S.; Yang, S.; Wang, S.; Tan, S.; Wang, P.; Lin, J.; Zhou, C.; and Zhou, J. 2023. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*.

Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; Dang, K.; Wang, P.; Wang, S.; Tang, J.; et al. 2025. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*.

Cao, J.; Ye, P.; Li, S.; Yu, C.; Tang, Y.; Lu, J.; and Chen, T. 2024. MADTP: Multimodal alignment-guided dynamic token pruning for accelerating vision-language transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15710–15719.

Chen, L.; Zhao, H.; Liu, T.; Bai, S.; Lin, J.; Zhou, C.; and Chang, B. 2024a. An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models. In *Proceeding of the European Conference on Computer Vision*, 19–35.

Chen, Y.; Wang, G.; Shang, J.; Cui, S.; Zhang, Z.; Liu, T.; Wang, S.; Sun, Y.; Yu, D.; and Wu, H. 2024b. NACL: A general and effective KV cache eviction framework for LLM at inference time. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 7913–7926.

Chu, X.; Qiao, L.; Zhang, X.; Xu, S.; Wei, F.; Yang, Y.; Sun, X.; Hu, Y.; Lin, X.; Zhang, B.; et al. 2024. MobileVLM v2: Faster and stronger baseline for vision language model. *arXiv preprint arXiv:2402.03766*.

Dettmers, T.; Lewis, M.; Belkada, Y.; and Zettlemoyer, L. 2022. GPT3.INT8 (): 8-bit matrix multiplication for transformers at scale. *In Proceeding of the Advances in Neural Information Processing Systems*, 35: 30318–30332.

Dingjie, S.; Chen, S.; Chen, G. H.; Yu, F.; Wan, X.; and Wang, B. 2024. Milebench: Benchmarking MLLMs in long context. In *Proceeding of the Conference on Language Modeling*.

Duan, H.; Yang, J.; Qiao, Y.; Fang, X.; Chen, L.; Liu, Y.; Dong, X.; Zang, Y.; Zhang, P.; Wang, J.; et al. 2024. VLMEvalkit: An open-source toolkit for evaluating large multi-modality models. In *Proceedings of the ACM International Conference on Multimedia*, 11198–11201.

Frantar, E.; Ashkboos, S.; Hoefler, T.; and Alistarh, D. 2023. GPTQ: Accurate post-training quantization of generative pretrained transformers. *In Proceeding of The International Conference on Learning Representations*.

Huang, W.; Zhai, Z.; Shen, Y.; Cao, S.; Zhao, F.; Xu, X.; Ye, Z.; and Lin, S. 2025. Dynamic-LLaVA: Efficient multimodal large language models via dynamic vision-language context sparsification. In *Proceeding of The International Conference on Learning Representations*.

Li, B.; Zhang, Y.; Guo, D.; Zhang, R.; Li, F.; Zhang, H.; Zhang, K.; Zhang, P.; Li, Y.; Liu, Z.; et al. 2025. LLaVA-OneVision: Easy visual task transfer. *Transactions on Machine Learning Research*, 2025.

Li, Y.; Huang, Y.; Yang, B.; Venkitesh, B.; Locatelli, A.; Ye, H.; Cai, T.; Lewis, P.; and Chen, D. 2024a. SnapKV: LLM knows what you are looking for before generation. *In Proceedings of the Advances in Neural Information Processing Systems*, 37: 22947–22970.

Li, Z.; Yang, B.; Liu, Q.; Ma, Z.; Zhang, S.; Yang, J.; Sun, Y.; Liu, Y.; and Bai, X. 2024b. Monkey: Image resolution and text label are important things for large multi-modal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 26763–26773.

Liu, H.; Li, C.; Li, Y.; Li, B.; Zhang, Y.; Shen, S.; and Lee, Y. J. 2024a. Llava-next: Improved reasoning, OCR, and world knowledge. *https://llava-vl. github. io/blog/2024-01-30-llava-next*.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023. Visual instruction tuning. In *Proceedings of the Advances in Neural Information Processing Systems*.

Liu, Y.; Li, Z.; Huang, M.; Yang, B.; Yu, W.; Li, C.; Yin, X.-C.; Liu, C.-L.; Jin, L.; and Bai, X. 2024b. OCRBench: On the hidden mystery of OCR in large multimodal models. *Science China Information Sciences*, 67(12): 220102.

Lu, P.; Bansal, H.; Xia, T.; Liu, J.; Li, C.; Hajishirzi, H.; Cheng, H.; Chang, K.-W.; Galley, M.; and Gao, J. 2024. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *Proceeding of The International Conference on Learning Representations*.

Ma, X.; Fang, G.; and Wang, X. 2023. LLM-pruner: On the structural pruning of large language models. *In Proceeding of the Advances in Neural Information Processing Systems*, 36: 21702–21720.

Pei, X.; Huang, T.; and Xu, C. 2024. Cross-self KV cache pruning for efficient vision-language inference. *arXiv preprint arXiv:2412.04652*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning*, 8748–8763.

Shang, Y.; Cai, M.; Xu, B.; Lee, Y. J.; and Yan, Y. 2024. LLaVA-prumerge: Adaptive token reduction for efficient large multimodal models. *arXiv preprint arXiv:2403.15388*.

Sun, M.; Liu, Z.; Bair, A.; and Kolter, J. Z. 2024. A simple and effective pruning approach for large language models. In *Proceeding of The International Conference on Learning Representations*.

Team, K.; Du, A.; Yin, B.; Xing, B.; Qu, B.; Wang, B.; Chen, C.; Zhang, C.; Du, C.; Wei, C.; et al. 2025. Kimi-VL technical report. *arXiv preprint arXiv:2504.07491*.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023a. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023b. LLaMA 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(11): 2579–2605.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L. u.; and Polosukhin, I. 2017. Attention is All you Need. In *Proceeding of the Advances in Neural Information Processing Systems*.

Wan, Z.; Shen, H.; Wang, X.; Liu, C.; Mai, Z.; and Zhang, M. 2025. MEDA: Dynamic KV cache allocation for efficient multimodal long-context inference. In *Proceedings of the Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 2485–2497.

Wan, Z.; Wu, Z.; Liu, C.; Huang, J.; Zhu, Z.; Jin, P.; Wang, L.; and Yuan, L. 2024. LOOK-M: Look-once optimization in KV cache for efficient multimodal long-context inference. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 4065–4078.

Wang, X.; Alam, S.; Wan, Z.; Shen, H.; and Zhang, M. 2025a. SVD-LLM v2: Optimizing singular value truncation for large language model compression. In *Proceedings of the Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 4287–4296.

Wang, X.; Zheng, Y.; Wan, Z.; and Zhang, M. 2025b. SVD-LLM: Truncation-aware singular value decomposition for large language model compression. In *Proceeding of the International Conference on Learning Representations*.

Yu, W.; Yang, Z.; Li, L.; Wang, J.; Lin, K.; Liu, Z.; Wang, X.; and Wang, L. 2024. MM-Vet: Evaluating large multimodal models for integrated capabilities. In *Proceeding of the International Conference on Machine Learning*.

Zhang, S.; Roller, S.; Goyal, N.; Artetxe, M.; Chen, M.; Chen, S.; Dewan, C.; Diab, M.; Li, X.; Lin, X. V.; et al. 2022. OPT: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Zhang, Z.; Sheng, Y.; Zhou, T.; Chen, T.; Zheng, L.; Cai, R.; Song, Z.; Tian, Y.; Ré, C.; Barrett, C.; et al. 2023. H2O: Heavy-hitter oracle for efficient generative inference of large language models. *In proceedings of the Advances in Neural Information Processing Systems*, 36: 34661–34710.