H^2SVC: Head-aware Heterogeneous Streaming Video Cache for Online Video Understanding

Han Li[1], Xinyi Zhang[1], Wenrui Dai[1*], Yaoming Wang[1*], Xinyu Peng[1], Fan He[2], Hang Xu[2], Ziyang Zheng[1], Chenglin Li[1], Junni Zou[1], and Hongkai Xiong[1]

{qingshi9974,zhangxinyi_0406,daiwenrui,wang_yaoming}@sjtu.edu.cn
{xypeng9903,zhengziyang,lcl1985,zoujunni,xionghongkai}@sjtu.edu.cn

[1] Shanghai Jiao Tong University
[2] Yinwang Intelligent Technology Co. Ltd.

Abstract. Multimodal Large Language Models (MLLMs) suffer from severe memory bottlenecks in streaming video understanding due to accumulated Key-Value (KV) cache with the growth of video length. Existing training-free compression methods employ the same strategy across all attention heads in one layer, but ignore their diverse functional roles in temporal representation. In this paper, we reveal the inherent temporal heterogeneity of attention heads in MLLMs through systematic Spatio-temporal Attention Profiling (SAP), and discover that heads naturally specialize into instantaneous, short-term, and episodic roles agnostic to input video frames and their queries. Motivated by this finding, we propose the Head-aware Heterogeneous Streaming Video Cache (H^2SVC), a training-free KV cache compression framework for efficient streaming video understanding. H^2SVC features an offline routing strategy that exploits the distinct temporal behavior of heads in each layer and adapt their KV cache to dedicated memory banks tailored for varying temporal receptive fields. Furthermore, we develop Semantic Trajectory Curvature Eviction (STCE) for the long-term episodic bank to enhance semantic diversity under strict streaming budget. STCE models the Value states as a geometric trajectory to continuously evict predictable frames on smooth paths, while preserving sharp semantic inflection points. Extensive experiments on widely adopted streaming benchmarks demonstrate that H^2SVC achieves state-of-the-art performance, and is comparable or superior to full-context offline models with significantly reduced KV cache memory and accelerated inference speed.

Keywords: Streaming Video · Multimodal Large Language Models · KV Cache Compression · Attention Heterogeneity

1 Introduction

Multimodal Large Language Models (MLLMs) [7,11,13,14,28,37,38] have achieved remarkable progress in video understanding. However, it emerges as a fundamen-

* Corresponding author.

tal challenge to deploy MLLMs in the real-world streaming scenarios with continuously acquired video frames. Autoregressive KV cache is designed to avoid redundant computation, but increases GPU memory cost with the growth of video length and could fail in long-term applications like surveillance or sports broadcasting. For example, a ten-minute video at 1 frame per second (FPS) cannot be handled by most single modern GPU due to limited memory capacity.

Existing approaches address this memory bottleneck from two aspects. *Architectural methods* [3,24,30,32,32,33] reduce the number of tokens per frame or introduce specialized trained memory modules to aggregate historical context. However, they are not compatible with off-the-shelf frozen MLLMs, since they require costly fine-tuning and alter the original model architecture. *Compression methods* [5,9,22,35,36] restrict the cache size using first-in-first-out (FIFO) eviction, similarity-based merging, or prompt-guided retrieval. However, they ignore the fact that different heads could serve entirely different temporal roles and are limited by the same compression strategy for each attention head within a layer.

In this paper, we reveal that *attention heads in a MLLM inherently achieves streaming video understanding with different types of temporal behavior*. We perform a systematic Spatiotemporal Attention Profiling (SAP) across the stages of visual prefilling and text querying, and discover that attention heads naturally fall into three distinct categories: **i) instantaneous heads** that focus on the spatial details of current frame, **ii) short-term heads** that concentrate on recent temporal context, and **iii) episodic heads** that maintain a global receptive field over the entire history of streaming video. We emphasize that this specialization of heads is an *inherent* property of the pre-trained weights, and remains consistent across different videos and query inputs. The significantly varying temporal behavior of heads within a single layer suggests that existing methods yield sub-optimal layer-wise compression with uniform strategy.

Motivated by this finding, we propose the **Head-aware Heterogeneous Streaming Video Cache ($\mathbf{H^2SVC}$)** framework for training-free KV cache compression to allow streaming videos of infinite length. H^2SVC is fully compatible with hardware-efficient kernels such as Flash Attention [8] by entirely operating on raw Key and Value vectors without requiring the attention weight matrix. We first classify all the heads into the instantaneous, short-term and episodic categories using our profiling metrics in an offline manner without incurring extra compute cost in subsequent inference. Consequently, we design a specific KV cache compression strategy to each category of heads via dedicated memory banks. Particularly, we develop Online Semantic Trajectory Curvature Eviction (STCE) for the global memory bank of episodic heads to maintain a bounded yet semantically diverse episodic memory under the strict streaming budget. STCE models the Value state as a continuous geometric trajectory rather than traditional pairwise similarity to simultaneously evict predictable frames (with low curvature) and preserve sharp semantic turning points.

Our contributions are summarized as below.

- We reveal the temporal heterogeneity of attention heads in MLLMs for streaming video understanding through a systematic SAP, and demonstrate

the input-agnostic property of instantaneous, short-term and episodic heads in temporal behavior.
- We propose **H^{2}SVC** that exploits the temporal heterogeneity of attention heads and achieves head-aware KV cache compression in a training-free fashion. We adapt KV cache of heads of different temporal behavior to specific memory banks with offline Head-Aware Heterogeneous KV Routing for high-efficiency compression.
- We further develop Semantic Trajectory Curvature Eviction (STCE) for episodic heads with global memory bank. STCE resorts to a geometry-aware online algorithm that identifies and evicts predictable frames, while preserving key semantic events.
- Extensive experiments across widely adopted benchmarks (*e.g.* Streaming-Bench, OVO-Bench) and three prevailing backbones show that H^{2}SVC consistently outperforms existing training-free offline-to-online methods. Remarkably, H^{2}SVC achieves state-of-the-art performance and matches full-context offline models with reduced KV cache memory.

2 Related Work

Streaming and Online Video Understanding. Early efforts to adapt offline MLLMs for streaming video primarily focused on architectural and training-level innovations. TimeChat-Online [32] designs a Differential Token Drop module to eliminate redundant visual tokens. StreamForest [33] organizes streaming frames into persistent event-level tree structures, and StreamingVLM [30] aligns training and inference through an overlapped-chunk attention strategy. However, these methods ignore the inherent heterogeneity of different heads in perceiving the video stream. They require costly supervised fine-tuning on large-scale streaming data or uniformly apply the same temporal compression policy across all attention heads.

Sparse Attention in LLMs. To avoid the high cost of architectural modifications, sparse attention and KV cache eviction have been widely explored for handling long contexts in LLMs [16, 19, 29, 34, 39]. StreamingLLM [29] preserves initial "attention sinks" and recent tokens via sliding window attention, while H_2O [39] and SnapKV [16] dynamically evict tokens with low accumulated attention scores. Despite effective for pure texts, these token-level strategies cannot be easily transferred to streaming videos. Unlike discrete text tokens with relatively uniform semantic density, video frames exhibit high Spatio-temporal redundancy and continuous semantic transitions. Attention-guided eviction often struggles to distinguish genuine dynamic events from static background noise.

KV Cache Compression for Streaming Video. To address the unique challenges of video redundancy, a parallel line of research targets the KV cache bottleneck via video-specific training-free compression. These methods employ strategies like FIFO eviction [35], query-guided retrieval [5, 9, 22], dual-metric scoring [12], or layer-wise hierarchical budgets [36]. However, existing methods suffer from two shared critical limitations. First, they treat all attention

heads within a layer identically, and fail to exploit fine-grained head-wise temporal specialization. Second, they are incompatible with hardware-efficient kernel such as Flash Attention [8], since these methods rely on inference-time attention weights [22, 36] and have to materialize the full $\mathcal{O}(N^2)$ attention matrix. Our work addresses these limitations through offline head-aware KV routing and a geometry-aware eviction strategy that operates purely on the Value vectors.

3 Motivation: Temporal Heterogeneity of Attention Heads

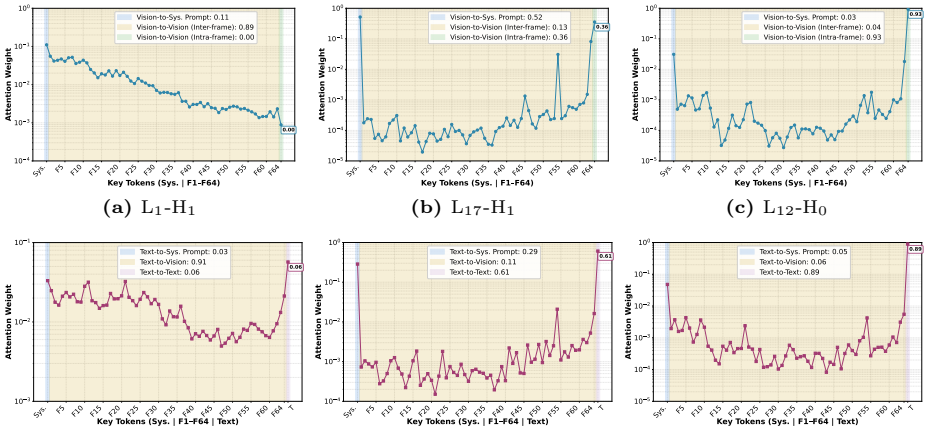
A fundamental assumption underlying existing streaming video KV cache methods is that all attention heads within a transformer layer share the same temporal behavior, and therefore, deserve the same compression, sliding, or retrieval strategy. We challenge this through two complementary analyses: a qualitative visualization on individual heads in Section 3.1, and a systematic quantitative profiling across the entire model in Section 3.2.

3.1 Head-Level Attention Visualization

We first analyze the attention distributions of LLaVA-Video-7B [38] on a 64-frame streaming video to understand how visual and text tokens attend to historical video frames. Specifically, we examine three representative attention heads across two stages: (1) **visual prefilling**, measuring how a newly arrived frame attends to past context, and (2) **text querying (including text prefilling and decoding)**, measuring how text tokens attend to the video context. As shown in Figure 1, these heads exhibit significantly distinct temporal behaviors:

- **Layer1-Head1** maintains a global temporal receptive field to extract long-term context. It allocates dominant attention (89%) to inter-frame tokens during visual prefilling, and remains broadly distributed across the entire video history during text querying.
- **Layer17-Head1** shows a strong recency bias. In both visual prefilling and text querying, attention concentrates heavily on the most recent frames, which are responsible for tracking recent motion dynamics.
- **Layer12-Head0** exhibits weak dependency on historical temporal context. In visual prefilling, attention is almost exclusively allocated to the intra-frame spatial tokens of the current frame (93%), and text querying also shows negligible weight over the visual history. Such heads are dedicated to extracting immediate semantics of current context rather than maintaining historical visual memory.

These observations reveal a clear division of roles in how the model processes streaming video: different attention heads naturally specialize in distinct temporal receptive fields.



(a) L_1-H_1 **(b)** L_{17}-H_1 **(c)** L_{12}-H_0

(d) L_1-H_1 **(e)** L_{17}-H_1 **(f)** L_{12}-H_0

Fig. 1: Attention behavior across different heads. L_i-H_j denotes the j-th head in the i-th LLM layer. Each point denotes the *average* attention weight from query tokens to a specific context segment (Sys.: system prompt; $\mathbf{F}_i$: the i-th video frame; $\mathbf{T}$: text tokens). The number in the legend indicates the total attention mass for that segment. **Top row (visual prefilling)** shows the average attention from the last frame as query to preceding context. **Bottom row (text querying)** shows the average attention from all text tokens (question and answer) as query to the preceding context.

3.2 Spatio-temporal Attention Profiling (SAP)

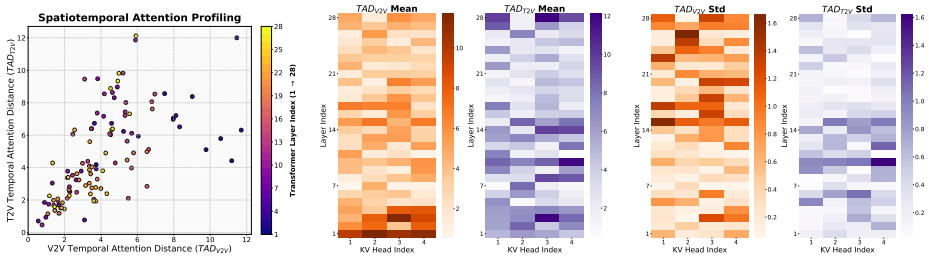
To systematically quantify the temporal heterogeneity observed in our motivational analysis, we conduct a comprehensive profiling experiment. We randomly sample a diverse set of videos (detailed in Appendix B) and prompt the model with a query: *"Please provide a detailed description of the video, focusing on the main subjects, their actions, the background scenes."*

During inference, we extract and aggregate the attention maps for all Key-Value (KV) heads[3] across the 28 transformer layers of LLaVA-Video-7B. To precisely measure the temporal receptive field of each head from these maps, we define two metrics: Vision-to-Vision Temporal Attention Distance (TAD_{V2V}) and Text-to-Vision Temporal Attention Distance (TAD_{T2V}).

Formally, let F_i be the set of visual tokens belonging to the i-th video frame, where $i \in \{1, 2, \ldots, N\}$ and N is the total number of frames. For a given query token q at a specific KV head h, we define $A^{(h)}_{q,F_j} = \sum_{k \in F_j} A^{(h)}_{q,k}$ as the aggregated attention weight directed towards the entire j-th frame. For models employing Grouped-Query Attention (GQA), the attention map for a KV head is computed as the average of attention maps from its associated group of Query heads.

The TAD_{V2V} quantifies the expected temporal look-back distance during the visual prefilling stage. It is formulated as the expectation over all queries

[3] LLaVA-Video employs Grouped-Query Attention (GQA) [1] with 4 KV heads per layer. Since KV cache compression operates on KV states, our profiling targets these KV heads rather than the query heads.



(a) (b) (c)

Fig. 2: TAD profiling of KV heads in LLaVA-Video-7B. (a) 2D scatter of TAD_{V2V} vs. TAD_{T2V} for all heads; colors indicate layer depth (1→28). (b) Mean TAD heatmap per head across diverse videos. (c) Standard deviation of TAD scores, showing each head's temporal preference is stable across different input videos.

across all frames, and further averaged over the entire sampled video set $\mathcal{V}$:

$$TAD_{V2V}^{(h)} = \mathbb{E}_{v\in\mathcal{V}}\left[\frac{1}{N_v}\sum_{i=1}^{N_v}\mathbb{E}_{q\in F_i}\left[\sum_{j=1}^{i} A_{q,F_j}^{(h)}\cdot(i-j)\right]\right] \tag{1}$$

Here, N_v is the total number of frames in video v, i is the index of the current query frame, and j is the index of the historical key frame. Intuitively, if TAD_{V2V} approaches 0, it indicates that the video frames almost exclusively attend to intra-frame spatial tokens ($j = i$), exhibiting no historical look-back. Conversely, a significantly larger TAD_{V2V} signifies a broader temporal receptive field, meaning the head consistently aggregate distant, global historical visual context across the temporal axis.

Similarly, the TAD_{T2V} evaluates the temporal distance that text query q_t attend to visual context, where the reference point for the temporal distance is the final frame N_v, and the text queries can attend to the entire historical visual sequence $j \in \{1, \ldots, N_v\}$:

$$TAD_{T2V}^{(h)} = \mathbb{E}_{v\in\mathcal{V}}\left[\mathbb{E}_{q_t}\left[\sum_{j=1}^{N_v} A_{q_t,F_j}^{(h)}\cdot(N_v-j)\right]\right] \tag{2}$$

Similarly, a small TAD_{T2V} implies that the text tokens focus on the most recent video frames, while a larger TAD_{T2V} demonstrates that the text tokens retrieve information from distant, global historical frames to formulate the response.

As illustrated in Figure 2, we visualize the distribution of TAD_{V2V} and TAD_{T2V} for all KV heads. We identify three critical findings that fundamentally challenge the conventional design of video KV caches:

– **Finding 1: Consistent Temporal Dynamics across Modalities.** The strong positive correlation between TAD_{V2V} and TAD_{T2V} indicates that heads preferring long-range context during visual prefilling maintain this global visual receptive field for text querying. This alignment reveals that visual and text tokens share highly consistent temporal attention patterns when processing historical visual context.

- **Finding 2: Significant Intra-Layer Head Heterogeneity.** Heads belonging to the same layer (identical colors in Figure 2(a)) are widely scattered across the TAD spectrum. This high intra-layer variance proves that traditional layer-wise cache management is inherently sub-optimal, necessitating a strictly fine-grained, *head-wise* cache management strategy.
- **Finding 3: Input-Agnostic Intrinsic Robustness.** Across diverse videos and prompts, the variance of TAD scores remains small (see Appendix B). This indicates that a head's spatio-temporal preference is not heavily conditioned on the input, but is an inherent property of the pre-trained weights.

Motivated by these findings, we propose the **Head-aware Heterogeneous Streaming Video Cache (H²SVC)** framework to leverage this inherent temporal heterogeneity for efficient long-video understanding.

4 Proposed Framework for KV Cache Compression

The proposed H²SVC framework enables infinite video streaming with a strictly bounded memory budget through two tightly coupled components. First, we categorize each head into different types according to computed TAD and implement an offline static routing strategy that assigns each head's KV cache to a dedicated memory bank. Second, to maintain the memory bound during online streaming, we introduce a online geometry-aware eviction strategy, named **Semantic Trajectory Curvature Eviction (STCE)**, optimized specifically for the long-term episodic bank.

4.1 Offline Head-Aware Heterogeneous KV Routing

Phase 1: Offline Head Classification. We normalize both TAD_{V2V} and TAD_{T2V} computed during profiling stage to a $[0, 1]$ range across all H attention heads. We then calculate a unified Temporal Receptive Score $S^{(h)}$ for each KV head h by averaging the two normalized values, where $\mathcal{N}(\cdot)$ is min-max normalization. This score reflects how much the head relies on historical visual context:

$$S^{(h)} = \frac{1}{2}\left(\mathcal{N}(TAD^{(h)}_{V2V}) + \mathcal{N}(TAD^{(h)}_{T2V})\right) \tag{3}$$

Next, we sort all heads by their $S^{(h)}$ scores and divide them into three distinct categories. NOTE THAT this classification is performed only once offline prior to deployment, yielding a static Head Classification Map C_h:

- **Instantaneous Heads** (Bottom 10%): Heads with the lowest $S^{(h)}$, focusing primarily on the immediate, ultra-short temporal context.
- **Short-term Heads** (Middle 40%): Heads with moderate $S^{(h)}$, tracking recent temporal context.
- **Episodic Heads** (Top 50%): Heads with the highest $S^{(h)}$, maintaining a global receptive field over the entire video history.

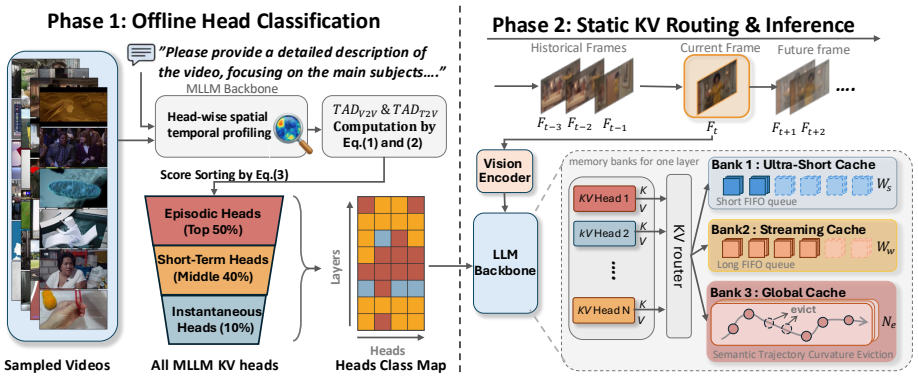


Fig. 3: Overview of the H^{2}SVC framework. Left: We propose a spatialtemporal profiling to classify all KV heads into instantaneous, short-term, and episodic heads. Right: During inference, we route the KV cache of each head into banks according to its head type, including ultra-short cache, streaming cache, and global cache.

Phase 2: Static KV Routing. During inference, each head's KV states are statically routed to specific heterogeneous memory bank according to the pre-computed map C_h with corresponding compression strategy:

- **Instantaneous Bank:** Heads classified as Instantaneous are routed to an Ultra-Short Cache. This cache only keeps a small FIFO sliding window of W_s frames, saving memory for heads that exclusively focus on immediate spatial details.
- **Short-term Bank:** Heads classified as Short-term are routed to a Streaming Cache. It maintains a standard FIFO sliding window of W_w frames to track recent motion and short-term changes.
- **Episodic Bank:** Heads classified as Episodic require global context and are routed to a Global Cache with a budget of N_e frames. Instead of a simple sliding window, this cache uses our proposed Semantic Trajectory Curvature Eviction to continually drop redundant frames, keeping important long-range events indefinitely.

In this way, during both visual prefilling and text querying, the Multi-Head Attention mechanism naturally accommodates the heterogeneous cache lengths. Specifically, for each attention head h, its corresponding query states $\mathbf{Q}^{(h)}$ interact with the updated KV cache $\mathbf{K}^{(h)}_{cache}$ and $\mathbf{V}^{(h)}_{cache}$ maintained by its assigned memory bank. Then, the outputs from all heads are simply concatenated along the head dimension as in standard transformer architectures, which ensures that our heterogeneous routing introduces no structural modification to the attention computation itself.

4.2 Online Semantic Trajectory Curvature Eviction (STCE)

For the Episodic heads, we restrict the Global Memory bank with a limit of N_e frames and define an importance score $I(\cdot)$ for each cached frame to guide

eviction. When the bank is full, the frame with the lowest importance score $I(\cdot)$ is evicted upon the arrival of each incoming frame. A straightforward baseline is *Uniform Eviction* that discards frames to maintain equal temporal spacing. However, it is content-agnostic and cannot distinguish semantically rich events from redundant transitions. A more informed strategy is *Similarity Eviction* that removes a frame when it closely resembles its neighbors, but could fail under smooth continuous motions. A frame may differ from its neighbors yet remain highly predictable, as it lies on a straight interpolatable path between them.

To address this, we model the sequence of cached Value states as a geometric trajectory in the latent feature space. A frame lying on a smooth path between its temporal neighbors is semantically predictable and thus redundant. In contrast, a frame at a sharp turn in the trajectory marks a semantic inflection point or event boundary. We therefore use the *geometric curvature* at each frame's position to measure its informational value.

Crucially, we compute this trajectory strictly using the **Value states (V)** rather than the Key states (K), as Rotary Position Embeddings (RoPE) [27] encodes absolute position into K, introducing positional bias that distorts geometric distances and confounds semantic similarity. Let $\{F_j\}_{j=1}^{N_e}$ denote the frames currently stored in the Global Memory bank, ordered by arrival time, where j is its position index in the memory bank. For frame F_j, let V_j denote its frame-level Value representation, we define $\overrightarrow{v}_{in} = V_j - V_{j-1}$ and $\overrightarrow{v}_{out} = V_{j+1} - V_j$ be the semantic velocity in Value space, where F_{j-1} and F_{j+1} are temporally adjacent frames in the cache. The importance score $I(j)$ is the product of three terms:

$$I_{curve}(j) = \underbrace{(1 - \text{CosSim}(\overrightarrow{v}_{in}, \overrightarrow{v}_{out}))}_{\text{Angular Curvature}} \cdot \underbrace{\min(||\overrightarrow{v}_{in}||_2, ||\overrightarrow{v}_{out}||_2)}_{\text{Magnitude Gating}} \cdot \underbrace{\log(t_{j+1} - t_{j-1})}_{\text{Temporal Weight}}. \tag{4}$$

1. **Angular Curvature:** $1 - \text{CosSim}(\overrightarrow{v}_{in}, \overrightarrow{v}_{out})$. This term measures the directional change of trajectory. A near-zero value indicates a smooth, predictable transition, whereas a larger value signals a sharp semantic turning.
2. **Magnitude Gating:** $\min(\|\overrightarrow{v}_{in}\|_2, \|\overrightarrow{v}_{out}\|_2)$. In static scenes, small feature variations are mostly noise and can produce unstable angle estimates. This term suppresses false high-curvature scores in low-motion regions.
3. **Temporal Weight:** $\log(t_{j+1} - t_{j-1})$, where t_j denotes the original frame index of F_j in the video stream. This term favors frames that span larger temporal intervals in the cache, preventing the bank from being dominated by densely sampled adjacent frames.

A frame receives a low score if it is semantically predictable (low curvature), lies in a static region (low magnitude), or covers only a small temporal span. Importantly, evicting a frame F_j only invalidates the scores of its two neighbors F_{j-1} and F_{j+1}, so we recompute only those two scores rather than rescoring the entire bank. This local update keeps the per-frame overhead constant regardless of bank size, ensuring the Global Cache scales efficiently to infinitely long streaming videos. Algorithm 1 in Appendix C describes the complete procedure.

5 Experiments

5.1 Experimental Setup

Benchmarks. We evaluate our method across two categories of benchmarks to comprehensively assess its capabilities. For **Online Benchmarks**, we use StreamingBench [18], OVO-Bench [23], ODV-Bench [33], and RVS [35] to test real-time video understanding, and streaming temporal reasoning under strict memory constraints. For **Offline Benchmarks**, we evaluate on Video-MME [10], MLVU [40], and EgoSchema [21] to verify that our streaming cache compression incurs minimal performance degradation on the model's general video comprehension abilities compared to full-cache offline inference.

Baselines and Backbone Models. H^{2}SVC is a plug-and-play, training-free method that can be applied to any off-the-shelf MLLM. We report results across three backbone families: LLaVA-OneVision-7B [13], LLaVA-Video-7B [38], and Qwen2.5-VL-7B [2]. We compare against (i) open-source offline MLLMs, including Video-LLaVA [17], VideoChat2 [15], LongVA [37], Kangaroo [20], and VideoXL [25], (ii) open-source online MLLMs, including VideoLLM-Online [3], TimeChat-Online [32], DiSpider [24], StreamForest [33], and (iii) training-free offline-to-online methods, including ReKV [9], LiveVLM [22], StreamKV [5], Flash-VStream [35], InfiniPot-V [12], StreamMem [31], and StreamingTOM [4].

Implementation Details. All experiments use the default cache budget **M**: $(N_e, W_w, W_s) = (64, 16, 4)$ frames per head unless noted otherwise. During online inference, incoming video streams are processed via frame-by-frame encoding and prefilling. Head profiling and classification is performed offline on 50 randomly sampled videos using the combined TAD score $S^{(h)}$; the resulting classification map is then frozen for all inference runs. All models run on NVIDIA H800 GPU. No task-specific fine-tuning is applied at any stage.

5.2 Main Results

Table 1 reports performance across the online benchmarks. First, H^{2}SVC consistently outperforms other training-free offline-to-online methods across diverse backbones and benchmarks. On LLaVA-OV-7B, H^{2}SVC achieves 73.9 on StreamingBench (+1.0 over LiveVLM, +5.1 over StreamKV) and 61.0 on OVO-Bench, and even outperforms the dedicated streaming model StreamForest (56.61). Second, H^{2}SVC consistently surpasses full-context offline baselines on the online benchmarks, such as Backward subtest of OVO-Bench, where long-range episodic memory is critical, demonstrating that the Semantic Curvature Eviction selectively preserves semantically pivotal frames. Third, applying H^{2}SVC to the stronger LLaVA-Video-7B yields the highest overall numbers (77.5 / 62.6), confirming that head-aware KV routing scales effectively with the base model. Table 3 further validates these findings on the RVS benchmark; H^{2}SVC outperforms all comparisons on both Ego and Movie subsets.

Table 2 evaluates our streaming compression method on the offline benchmarks. Our H^{2}SVC shows comparable or even improved performance compared

Table 1: Performance comparison on online benchmarks. ‡: reproduced results.

Model	Size	#Frames	StreamingBench	OVO-Bench			ODV-bench
			Real-Time	Real-Time	Backward	Avg.	Avg.
Human	-	-	91.5	93.2	92.3	92.8	91.4
Open-source Offline MLLMs							
Video-LLaMA2 [7]	7B	32	49.5	-	-	-	-
LongVA [37]	7B	128	60.0	-	-	-	50.2
Qwen2-VL [28]	7B	64	69.0	60.7	48.6	54.6	55.6
InternVL2 [6]	8B	16	63.7	60.7	44.0	52.4	-
MiniCPM-V2.6 [11]	8B	32	67.4	-	-	-	49.8
Open-source Online MLLMs							
Flash-VStream [35]	7B	-	23.2	29.9	25.4	27.6	35.7
VideoLLM-Online [3]	8B	2 fps	36.0	20.8	17.7	19.3	-
Dispider [24]	7B	1 fps	67.6	54.6	36.1	45.3	45.2
TimeChat-Online [32]	7B	1 fps	75.4	61.9	41.7	51.8	-
StreamForest [33]	7B	1 fps	77.3	61.2	52.0	56.6	59.9
Training-free Offline-to-Online Methods							
LLaVA-OV [13]	7B	32	71.1	61.5	43.4	52.5	51.3
+ ReKV [9]	-	0.5 fps	69.1	64.4	46.7	55.6	50.9‡
+ LiveVLM [22]	-	0.5 fps	72.9	-	-	53.5‡	50.8‡
+ StreamKV [5]	-	0.5 fps	68.8	-	-	-	-
+ H²SVC	-	0.5 fps	73.9	65.7	56.3	61.0	52.4
LLaVA-Video [38]	7B	64	75.4‡	63.0‡	52.7‡	57.8‡	52.6‡
+ ReKV [9]	-	1 fps	72.9‡	-	-	57.1‡	52.1‡
+ LiveVLM [22]	-	1 fps	76.9‡	-	-	59.0‡	51.9‡
+ H²SVC	-	1 fps	77.5	67.2	58.0	62.6	53.1
Qwen2.5-VL [2]	7B	1 fps	72.6‡	64.7‡	48.3‡	56.5‡	57.4‡
+ ReKV [9]	-	1 fps	68.1‡	-	-	53.4‡	54.9‡
+ LiveVLM [22]	-	1 fps	73.2‡	-	-	56.3‡	57.3‡
+ InfiniPot-V [35]	-	1 fps	76.4	65.9	47.6	56.8	-
+ H²SVC	-	1 fps	77.0	67.7	52.8	60.3	58.7

to the full-cache offline baselines. For example, LLaVA-OV-7B + H²SVC achieves 65.1 on MLVU-dev compared to the offline baseline's 64.7. This suggests that our head-aware KV routing and semantic eviction not only save memory but also act as an effective noise filter, which removes useless temporal redundancy that might confuse the attention mechanism during long video tasks.

5.3 Ablation Study

To study the effect of each design component in H²SVC, we conduct ablation experiments using LLaVA-Video-7B on the Video-MME and OVO-Bench benchmarks. Results are detailed in Table 4.

(a) Effect of Different Classification Granularity. Head-wise classification and routing significantly outperforms

Table 3: Performance on RVS-Ego and RVS-Movie. †: upper bound.

Method	RVS-Ego		RVS-Movie	
	Acc	Sc.	Acc	Sc.
LLaVA-OV-7B [13]	56.2	3.7	43.0	3.3
+ ReKV† [9]	63.7	4.0	54.4	3.6
+ ReKV w/o off. [9]	55.8	3.3	50.8	3.4
+ Flash-VStream [35]	57.0	**4.0**	53.1	3.3
+ InfiniPot-V [12]	57.9	3.5	51.4	3.5
+ StreamMem [31]	57.6	3.8	52.7	3.4
+ StreamingTOM [4]	58.3	3.9	53.2	3.5
+ H²SVC	**58.9**	**4.0**	**54.2**	**3.6**

Table 2: Performance comparison on offline benchmarks. ‡: reproduced results.

Model	Size	#Frames	EgoSchema	MLVU-dev	VideoMME (w/o. sub)		
					Medium	Long	All
Open-source Offline Video LLMs							
Video-LLaVA-7B [17]	7B	8	38.4	47.3	38.0	36.2	39.9
VideoChat2-7B [15]	7B	16	54.4	47.9	37.0	33.2	39.5
LongVA [37]	7B	128	–	56.3	50.4	46.2	52.6
Kangaroo [20]	8B	64	–	61.0	55.3	46.6	56.0
VideoXL [25]	7B	128	–	64.9	53.2	49.2	55.5
Open-source Online Video LLMs							
MovieChat [26]	7B	2048	53.5	25.8	–	33.4	38.2
Dispider [24]	7B	1 fps	55.6	61.7	53.7	49.7	56.5
StreamForest [33]	7B	1 fps	–	–	–	–	61.4
Training-free Offline-to-Online Methods							
LLaVA-OV [13]	7B	32	60.7	64.7	57.6	47.4	58.4
+ H^2SVC	-	0.5 fps	62.1	65.1	56.9	48.6	59.0
LLaVA-Video‡ [38]	7B	64	58.0	67.5	61.6	52.3	63.3
+ H^2SVC	-	1 fps	59.0	67.0	60.7	53.4	63.0
Qwen2.5-VL‡ [2]	7B	1 fps	60.4	64.7	61.8	52.9	63.2
+ H^2SVC	-	1 fps	60.9	65.7	60.4	53.1	62.7

layer-wise routing (+2.5 on Video-MME, +0.7 on OVO-Bench). For layer-wise baseline, we average the TAD scores of all heads within a layer and assign every head in that layer to the same memory based on this single value. This forces heads with distinct temporal behaviors into a shared cache stream, improperly conflating fine-grained head-level specialization. The result validates Finding 2 (Section 3.2): the high intra-layer TAD variance renders layer-level assignment suboptimal.

(b) Effect of Head Ratio for Each Type. We ablate the ratio of heads assigned to each type. Using only Instantaneous heads degrades Video-MME to 49.1, as long-range context is entirely discarded; using only Episodic heads drops OVO-Bench to 59.5, losing fine-grained short-term cues. While our default setting achieves best performance. These results confirm that the three banks are both effective and complementary in the streaming video understanding.

(c) Effect of Memory Bank Budget. We compare three budget levels (S, M, L) with increasing (N_e, W_w, W_s) values. Scaling from S to M yields consistent gains (+1.3 on Video-MME, +2.7 on OVO-Bench). Further scaling to L provides gains on Video-MME (+1.5) but degrades OVO-Bench by 1.1. To balance accuracy-memory trade-off, we set the $(N_e, W_w, W_s) = (64, 16, 4)$ as default.

(d) Ablation of Curvature Eviction Components. We ablate each component of our STCE. As baselines, *Uniform Eviction* scores each frame by its temporal gap $I(j) = t_{j+1} - t_{j-1}$, retaining uniformly spaced frames; *Similarity Eviction* scores by its similarity to the preceding frame $I(j) = \mathrm{CosSim}(V_j, V_{j-1})$, retaining frames that differ most from their predecessor. On Video-MME, both

Table 4: Comprehensive Ablation Studies on Video-MME and OVO-Bench. Avg. KV size denotes the average length of visual KV cache retained per layer (assuming 182 tokens per frame for LLaVA-Video-7B). Default settings of our methods are marked with gray background.

Ablation / Setting	Avg. KV size	Video-MME (w/o. sub)			OVO-Bench		
		Medium	Long	All	Real-Time	Backward	Avg
(a) Classification Granularity							
Layer-wise	7K	58.8	52.0	60.5	66.1	57.7	61.9
Head-wise (Ours)	7K	**60.7**	**53.4**	**63.0**	**67.2**	**58.0**	**62.6**
(b) Head Ratio of Each Type (*Instantaneous* : *Short-term* : *Episodic*)							
1.00 : 0.00 : 0.00 (All Instantaneous)	0.7K	49.0	45.7	49.1	**70.1**	54.5	62.3
0.00 : 1.00 : 0.00 (All Short-term)	3K	51.6	48.7	54.0	68.0	57.1	62.6
0.00 : 0.00 : 1.00 (All Episodic)	12K	**61.8**	53.9	**64.0**	63.6	55.5	59.5
0.00 : 0.50 : 0.50	7K	60.8	52.7	62.8	65.9	56.7	61.3
0.33 : 0.33 : 0.33	5K	53.3	50.4	55.6	**68.3**	57.3	**62.8**
0.20 : 0.40 : 0.40	6K	54.4	51.4	57.0	66.9	**58.3**	62.6
0.10 : 0.40 : 0.50 (Ours)	7K	60.7	**53.4**	63.0	67.2	58.0	62.6
Full Context (Offline Baseline)	12K	61.6	52.3	63.3	63.0	52.7	57.8
(c) Memory Bank Budget (N_e, W_w, W_s)							
Budget S: (32, 8, 2)	3.5K	58.3	52.9	61.7	64.5	55.3	59.9
Budget M: (64, 16, 4) (Ours)	7K	60.7	53.4	63.0	**67.2**	**58.0**	**62.6**
Budget L: (128, 32, 8)	14K	**63.0**	**54.4**	**64.5**	65.5	57.4	61.5
(d) Curvature Eviction Components							
Uniform Eviction	7K	60.9	51.7	61.7	65.7	57.5	61.6
Semantic Eviction	7K	60.5	52.6	62.0	66.1	56.3	61.2
STCE w/ Key-based Trajectory	7K	60.4	53.1	62.7	66.5	56.7	62.1
STCE w/o Magnitude Gating	7K	60.2	52.7	62.6	65.5	57.3	61.4
STCE w/o Time Weight	7K	60.3	52.8	62.7	**68.0**	57.0	62.5
STCE (Ours)	7K	**60.7**	**53.4**	**63.0**	67.2	**58.0**	**62.6**
(e) Head Classification Metric.							
TAD_{V2V} only	7K	60.1	**53.7**	62.2	**68.2**	57.8	**63.3**
TAD_{T2V} only	7K	59.3	53.3	61.2	67.1	57.7	62.4
Combined Score $S^{(h)}$ (Ours)	7K	**60.7**	53.4	**63.0**	67.2	**58.0**	62.6

baseline underperform our STCE (61.6 and 61.2 vs. 62.6), confirming the superiority of our trajectory curvature-based scoring. Replacing Value-based curvature with Key-based (62.1) shows that RoPE in Key states distorts redundant frame eviction. Removing the Magnitude Gating term (61.4) or the Temporal Weight term (62.5) each degrade the performance, indicating that all components are necessary for suppressing static noise and maintaining temporal diversity.

(e) Effect of Head Classification Metric. We compare three variants of the TAD-based routing metric. Using only TAD_{V2V} (62.2 on Video-MME) ignores head behavior during text decoding, while using only TAD_{T2V} (61.2 on Video-MME, 62.4 on OVO-Bench) misses the visual aggregation patterns established during visual prefilling. The combined score $S^{(h)}$ achieves the best performance on both benchmarks, confirming that jointly modeling both stages is essential for accurate head classification.

5.4 Efficiency Analysis

To assess the practical deployment value of H^{2}SVC, we evaluate its efficiency from two perspectives: performance under constrained KV cache budgets, and memory and throughput scaling across different video lengths. All experiments use Qwen2.5-VL-7B with a fixed token budget of 256 tokens per frame.

Performance under Different KV Budgets. Table 5 compares H^{2}SVC with Qwen2.5-VL under varying KV budgets. For Qwen2.5-VL, the budget is controlled by setting an upper bound on the number of input frames; for H^{2}SVC, it refers to the average KV cache capacity across all heads. First, under the same tight budget of 5K, H^{2}SVC outperforms Qwen2.5-VL by +5.9 on OVO-Bench (59.5 vs. 53.6). Second, even when the budget is reduced by 90% relative to Qwen2.5-VL (10K vs. 100K), H^{2}SVC achieves higher OVO-Bench (61.1 vs. 57.5, +3.6) and nearly identical Video-MME (62.5 vs. 63.2, −0.7), showing its strong streaming understanding performance.

Table 5: Performance under varying KV cache/frame budgets (Qwen2.5-VL-7B).

Method	KV Budget	Video-MME (w/o sub)			OVO-Bench		
		Middle	Long	All	Real-Time	Backward	Avg.
Qwen2.5-VL	100K / 400 frames	61.3	52.7	63.2	63.8	51.2	57.5
	10K / 40 frames	60.9	51.2	62.8	61.8	48.0	54.9
	5K / 20 frames	57.3	48.9	60.6	61.1	46.0	53.6
+ H^{2}SVC (Ours)	10K / $(N_e, W_w, W_s) = (64, 16, 4)$	**60.2**	**52.7**	**62.5**	**67.8**	**54.4**	**61.1**
	5K / $(N_e, W_w, W_s) = (32, 8, 2)$	**58.0**	**50.1**	**61.3**	**67.7**	**51.3**	**59.5**

Efficiency and Scalability. As shown in Table 6, H^{2}SVC achieves consistent efficiency across all video lengths, while Full KV degrades rapidly as the length grows. (i) H^{2}SVC remains nearly constant for VRAM (17–19 GB), whereas Full KV grows linearly (17.8→76.2 GB), yielding a 4.0× reduction for 512 seconds streaming video. (ii) H^{2}SVC stays stable for visual refill at ∼20.5 FPS(frames processed per second), whereas Full KV drops to 12.4 FPS. (iii) Decoding throughput also remains steady at ∼42 tok/sec for H^{2}SVC , compared to Full KV's decline to 26.8 tok/sec. This result confirm the practicality for long-horizon streaming video understanding of our method.

Table 6: Efficiency comparison across different video lengths (Qwen2.5-VL-7B, single H800 GPU). For H^{2}SVC rows, red / green subscripts denote gains / drops vs. the Full KV baseline at the same video length.

Method	Length (sec)	VRAM (GB) ↓	FPS (frame/sec) ↑	Throughput (tok/sec) ↑
Full KV	64	17.8	22.9	48.9
H^{2}SVC	64	16.7 (-6.2%)	21.8 (-4.8%)	42.3 (-13.5%)
Full KV	128	20.5	20.5	43.2
H^{2}SVC	128	17.1 (-16.6%)	20.6 (+0.5%)	41.4 (-4.2%)
Full KV	256	38.6	17.6	36.4
H^{2}SVC	256	17.8 (-53.9%)	20.4 (+15.9%)	42.4 (+16.5%)
Full KV	512	76.2	12.4	26.8
H^{2}SVC	512	19.0 (-75.1%)	20.5 (+65.3%)	42.7 (+59.3%)

6 Conclusion

We presented **H^{2}SVC**, a training-free KV cache compression framework for infinite streaming video understanding. By discovering and leveraging the inherent temporal heterogeneity of attention heads, H^{2}SVC assigns each head's cache to a dedicated memory bank and employs a Semantic Trajectory Curvature Eviction strategy. This approach achieves state-of-the-art performance and maintains robust long-range reasoning while reducing KV cache memory by over 50%, all while remaining fully compatible with Flash Attention.

Acknowledgements

This work was in part supported by the National Natural Science Foundation of China, under Grants 62125109, 62431017, 62320106003, 62371288, U24A20251, 62401357, in part by the National Key Research and Development Program of China under Grant 2025YFF0515600, in part by the Fundamental Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China (JYB2025XDXM611), and in part by Yinwang Intelligent Technology Research Project.

References

1. Ainslie, J., Lee-Thorp, J., De Jong, M., Zemlyanskiy, Y., Lebrón, F., Sanghai, S.: Gqa: Training generalized multi-query transformer models from multi-head checkpoints. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. pp. 4895–4901 (2023)
2. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., et al.: Qwen2. 5-vl technical report. arXiv preprint arXiv:2502.13923 (2025)
3. Chen, J., Lv, Z., Wu, S., Lin, K.Q., Song, C., Gao, D., Liu, J.W., Gao, Z., Mao, D., Shou, M.Z.: Videollm-online: Online video large language model for streaming video. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18407–18418 (2024)
4. Chen, X., Tao, K., Shao, K., Wang, H.: Streamingtom: Streaming token compression for efficient video understanding. arXiv preprint arXiv:2510.18269 (2025)
5. Chen, Y., Bai, X., Wang, Z., Bai, C., Dai, Y., Lu, M., Zhang, S.: Streamkv: Streaming video question-answering with segment-based kv cache retrieval and compression. arXiv preprint arXiv:2511.07278 (2025)
6. Chen, Z., Wang, W., Tian, H., Ye, S., Gao, Z., Cui, E., Tong, W., Hu, K., Luo, J., Ma, Z., et al.: How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. Science China Information Sciences **67**(12), 220101 (2024)
7. Cheng, Z., Leng, S., Zhang, H., Xin, Y., Li, X., Chen, G., Zhu, Y., Zhang, W., Luo, Z., Zhao, D., et al.: Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. arXiv preprint arXiv:2406.07476 (2024)
8. Dao, T., Fu, D., Ermon, S., Rudra, A., Ré, C.: Flashattention: Fast and memory-efficient exact attention with io-awareness. Advances in neural information processing systems **35**, 16344–16359 (2022)

9. Di, S., Yu, Z., Zhang, G., Li, H., Zhong, T., Cheng, H., Li, B., He, W., Shu, F., Jiang, H.: Streaming video question-answering with in-context video kv-cache retrieval. arXiv preprint arXiv:2503.00540 (2025)
10. Fu, C., Dai, Y., Luo, Y., Li, L., Ren, S., Zhang, R., Wang, Z., Zhou, C., Shen, Y., Zhang, M., et al.: Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 24108–24118 (2025)
11. Hu, S., Tu, Y., Han, X., He, C., Cui, G., Long, X., Zheng, Z., Fang, Y., Huang, Y., Zhao, W., et al.: Minicpm: Unveiling the potential of small language models with scalable training strategies. arXiv preprint arXiv:2404.06395 (2024)
12. Kim, M., Shim, K., Choi, J., Chang, S.: Infinipot-v: Memory-constrained kv cache compression for streaming video understanding. arXiv preprint arXiv:2506.15745 (2025)
13. Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., et al.: Llava-onevision: Easy visual task transfer. arXiv preprint arXiv:2408.03326 (2024)
14. Li, H., Peng, X., Wang, Y., Peng, Z., Chen, X., Weng, R., Wang, J., Cai, X., Dai, W., Xiong, H.: Onecat: Decoder-only auto-regressive model for unified understanding and generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 30235–30245 (June 2026)
15. Li, K., He, Y., Wang, Y., Li, Y., Wang, W., Luo, P., Wang, Y., Wang, L., Qiao, Y.: Videochat: Chat-centric video understanding. Science China Information Sciences **68**(10), 200102 (2025)
16. Li, Y., Huang, Y., Yang, B., Venkitesh, B., Locatelli, A., Ye, H., Cai, T., Lewis, P., Chen, D.: Snapkv: Llm knows what you are looking for before generation. Advances in Neural Information Processing Systems **37**, 22947–22970 (2024)
17. Lin, B., Ye, Y., Zhu, B., Cui, J., Ning, M., Jin, P., Yuan, L.: Video-llava: Learning united visual representation by alignment before projection. In: Proceedings of the 2024 conference on empirical methods in natural language processing. pp. 5971–5984 (2024)
18. Lin, J., Fang, Z., Chen, C., Wan, Z., Luo, F., Li, P., Liu, Y., Sun, M.: Streamingbench: Assessing the gap for mllms to achieve streaming video understanding. arXiv preprint arXiv:2411.03628 (2024)
19. Liu, A., et al.: DeepSeek-V3.2: Pushing the frontier of open large language models. arXiv preprint arXiv:2512.02556 (2025)
20. Liu, J., Wang, Y., Ma, H., Wu, X., Ma, X., Wei, X., Jiao, J., Wu, E., Hu, J.: Kangaroo: A powerful video-language model supporting long-context video input. arXiv preprint arXiv:2408.15542 (2024)
21. Mangalam, K., Akshulakov, R., Malik, J.: Egoschema: A diagnostic benchmark for very long-form video language understanding. Advances in Neural Information Processing Systems **36**, 46212–46244 (2023)
22. Ning, Z., Liu, G., Jin, Q., Ding, W., Guo, M., Zhao, J.: Livevlm: Efficient online video understanding via streaming-oriented kv cache and retrieval. arXiv preprint arXiv:2505.15269 (2025)
23. Niu, J., Li, Y., Miao, Z., Ge, C., Zhou, Y., He, Q., Dong, X., Duan, H., Ding, S., Qian, R., et al.: Ovo-bench: How far is your video-llms from real-world online video understanding? In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 18902–18913 (2025)
24. Qian, R., Ding, S., Dong, X., Zhang, P., Zang, Y., Cao, Y., Lin, D., Wang, J.: Dispider: Enabling video llms with active real-time interaction via disentangled

perception, decision, and reaction. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 24045–24055 (2025)
25. Shu, Y., Liu, Z., Zhang, P., Qin, M., Zhou, J., Liang, Z., Huang, T., Zhao, B.: Video-xl: Extra-long vision language model for hour-scale video understanding. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 26160–26169 (2025)
26. Song, E., Chai, W., Wang, G., Zhang, Y., Zhou, H., Wu, F., Chi, H., Guo, X., Ye, T., Zhang, Y., et al.: Moviechat: From dense token to sparse memory for long video understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18221–18232 (2024)
27. Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., Liu, Y.: Roformer: Enhanced transformer with rotary position embedding. Neurocomputing **568**, 127063 (2024)
28. Wang, P., Bai, S., Tan, S., Wang, S., Fan, Z., Bai, J., Chen, K., Liu, X., Wang, J., Ge, W., et al.: Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. arXiv preprint arXiv:2409.12191 (2024)
29. Xiao, G., Tian, Y., Chen, B., Han, S., Lewis, M.: Efficient streaming language models with attention sinks. arXiv preprint arXiv:2309.17453 (2023)
30. Xu, R., Xiao, G., Chen, Y., He, L., Peng, K., Lu, Y., Han, S.: Streamingvlm: Real-time understanding for infinite video streams. arXiv preprint arXiv:2510.09608 (2025)
31. Yang, Y., Zhao, Z., Shukla, S.N., Singh, A., Mishra, S.K., Zhang, L., Ren, M.: Streammem: Query-agnostic kv cache memory for streaming video understanding. arXiv preprint arXiv:2508.15717 (2025)
32. Yao, L., Li, Y., Wei, Y., Li, L., Ren, S., Liu, Y., Ouyang, K., Wang, L., Li, S., Li, S., et al.: Timechat-online: 80% visual tokens are naturally redundant in streaming videos. In: Proceedings of the 33rd ACM International Conference on Multimedia. pp. 10807–10816 (2025)
33. Zeng, X., Qiu, K., Zhang, Q., Li, X., Wang, J., Li, J., Yan, Z., Tian, K., Tian, M., Zhao, X., et al.: Streamforest: Efficient online video understanding with persistent event memory. arXiv preprint arXiv:2509.24871 (2025)
34. Zhang, C., Bai, Y., Li, J., Gui, A., Wang, K., Liu, F., Wu, G., Jiang, Y., Bu, D., Wei, L., Jing, H., Tang, H., Chen, X., Huang, X., Li, F., Weng, R., Qian, Y., Lu, Y., Sun, Y., Wang, J., Xie, Y., Cai, X.: Efficient context scaling with longcat zigzag attention (2026)
35. Zhang, H., Wang, Y., Tang, Y., Liu, Y., Feng, J., Dai, J., Jin, X.: Flash-vstream: Memory-based real-time understanding for long video streams. arXiv preprint arXiv:2406.08085 (2024)
36. Zhang, H., Yang, S., Fu, J., Ng, S.K., Qiu, X.: Hermes: Kv cache as hierarchical memory for efficient streaming video understanding. arXiv preprint arXiv:2601.14724 (2026)
37. Zhang, P., Zhang, K., Li, B., Zeng, G., Yang, J., Zhang, Y., Wang, Z., Tan, H., Li, C., Liu, Z.: Long context transfer from language to vision. arXiv preprint arXiv:2406.16852 (2024)
38. Zhang, Y., Wu, J., Li, W., Li, B., Ma, Z., Liu, Z., Li, C.: Llava-video: Video instruction tuning with synthetic data. arXiv preprint arXiv:2410.02713 (2024)
39. 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. Advances in Neural Information Processing Systems **36**, 34661–34710 (2023)

40. Zhou, J., Shu, Y., Zhao, B., Wu, B., Liang, Z., Xiao, S., Qin, M., Yang, X., Xiong, Y., Zhang, B., et al.: Mlvu: Benchmarking multi-task long video understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13691–13701 (2025)