STREAMBANK: UNIFIED GENERATION–MEMORY INFERENCE FOR EFFICIENT STREAMING VIDEO UNDERSTANDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Streaming video understanding requires Video Large Language Models (Video-LLMs) to generate timely responses while preserving relevant historical information from continuous video streams. However, autoregressive response generation incurs substantial decoding latency, while continuously growing visual–textual histories introduce increasing computation and memory costs. Even with speculative decoding, candidate generation often remains sequential, while existing memory policies typically rely on visual similarity or recency rather than prediction-relevant evidence to retain historical events. To address these limitations, we propose **StreamBank**, a unified inference framework that uses target-model verification to accelerate response generation and guide historical event retention. StreamBank comprises two complementary components. First, *Reuse-Augmented MTP Speculative Decoding* predicts multiple future tokens in parallel and reuses previously verified responses as candidates when temporal continuity permits, reducing drafting overhead while retaining target-model verification. Second, *Speculation-Guided Event Memory* repurposes verifier-gain and correction signals from the same verification process to guide event promotion, retention, and eviction within a fixed-capacity memory bank. Together, these components connect efficient generation with prediction-aware memory management through shared verification feedback. Experiments on ProactiveVideoQA, OVO-Bench, and StreamingBench show that StreamBank largely preserves streaming video understanding performance under substantial context compression while achieving up to a $2.01\times$ decoding-throughput speedup.

1 INTRODUCTION

Video Large Language Models (Video-LLMs) have demonstrated strong capabilities in video understanding (Lin et al., 2024; Bai et al., 2025). However, unlike conventional offline video understanding, streaming video understanding requires a model to causally process continuously arriving visual information without access to future content, maintain historical context, and decide when to respond under strict latency constraints (Qian et al., 2024). Existing methods determine response timing through token-based triggering (Chen et al., 2024; Xia et al., 2026), verification-based scheduling (Yang et al., 2026), or binary response gating (Wang et al., 2024b). Token-triggered and verification-based approaches typically execute a full-model forward pass at every streaming step to determine whether a response is required. In contrast, binary gates can bypass model computation at some silent moments (Ding et al., 2025). Nevertheless, these mechanisms primarily optimize *when* to respond. Once a response is triggered, the backbone must still decode tokens autoregressively while processing visual context that continues to grow with the stream. Consequently, supporting proactive interaction alone does not guarantee efficient real-time inference over continuous video streams.

Computationally, streaming video understanding incurs overhead at two levels, as illustrated in Figure 1. At the generation level, once a response is triggered, the model decodes tokens autoregressively, causing serial forward computation and its associated latency to accumulate with response length. At the context level, the visual and textual history continually accumulates with the video stream, forcing the model to repeatedly process large amounts of redundant or task-irrelevant content

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

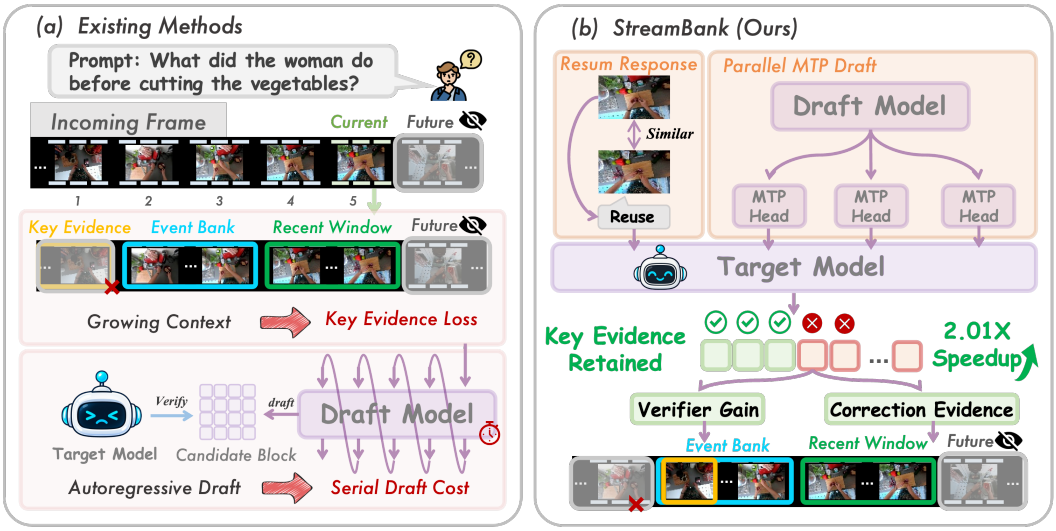

Figure 1: Inference comparison for streaming video understanding. (a) Existing methods process a growing visual context, risking key-evidence loss under memory compression, and generate candidate tokens autoregressively, incurring serial drafting costs. (b) StreamBank reuses previous target-verified responses under temporal continuity and drafts multiple tokens in parallel, while verifier-gain and correction evidence guide bounded event memory, preserving prediction-relevant evidence and achieving up to a 2.01× decoding speedup.

at every inference round and further increasing computation and storage costs. Speculative Decoding (SD) can mitigate the first source of overhead (Leviathan et al., 2023; Chen et al., 2023). It uses a lightweight draft model to generate a candidate block, which the target model verifies in a single parallel forward pass. Under a fixed verification context, longest-prefix acceptance preserves the target model's greedy output while reducing serial target-model calls. Meanwhile, we observe that the acceptance, rejection, and correction feedback produced during target verification in speculative decoding reflects the differences between the candidate block and the target model's predictions, providing task-relevant signals for estimating how historical information contributes to the current prediction. **Our Insight: Extend speculative decoding into a unified generation–memory inference process** *that reduces serial computation in response generation while reusing verification feedback to drive task-relevant historical event selection.*

Realizing this insight raises two key challenges. First, conventional speculative decoding uses either an independent lightweight draft model (Ji et al., 2025; Lin et al., 2025) or a shallow draft branch (Elhoushi et al., 2024; Liu et al., 2024) to generate a candidate block before target verification. Although this reduces serial target-model calls, the candidate block is commonly generated autoregressively, leaving multiple sequential candidate-generation steps in every speculative round. Parallel multi-token prediction (MTP) can remove this dependency within a response (Cai et al., 2024), but cannot exploit the cross-turn redundancy arising from highly similar responses across adjacent streaming turns. This leads to **Challenge 1**—Autoregressive Draft Generation Bottleneck: *How can a streaming Video-LLM reduce serial computation during candidate generation while further exploiting response redundancy across adjacent streaming turns?*

Second, existing streaming models commonly manage historical context by retaining the full history, using a fixed recent window, or constructing compressed event memory (Di et al., 2025; Shen et al., 2026; Zeng et al., 2025). Retaining the full history avoids forgetting, but repeatedly processes redundant or task-irrelevant content. Using only a recent window bounds computation, but discards early evidence. Event memory provides a compact alternative, yet existing compression policies commonly estimate event importance through heuristic signals such as temporal structure (Wang et al., 2025b), information density (Zhang et al., 2025a), or similarity-based retrieval (Ning et al., 2026), rather than through an event's actual contribution to model predictions. This motivates **Challenge 2**—Growing Context and Task-Relevant Event Management: *How can a streaming Video-*

LLM bound its historical context while identifying and retaining key events that contribute to subsequent response generation?

To address these challenges, we propose **StreamBank**, a unified generation–memory inference framework for streaming video understanding. The framework coordinates response generation with historical context management, reducing serial candidate-generation overhead while using target-verification feedback to dynamically identify and retain historical events that contribute to subsequent predictions. For **Challenge 1**, we introduce *Reuse-Augmented MTP Speculative Decoding*: multiple MTP heads generate a candidate block by predicting future tokens at different positions in parallel from the same shared representation, after which the unchanged target model verifies the candidate block in a single forward pass. When adjacent moments exhibit strong temporal continuity, we directly reuse the previous target-verified response or its prefix as the candidate block to exploit cross-turn redundancy. For **Challenge 2**, we introduce *Speculation-Guided Event Memory*, which maintains a fixed-capacity event bank and uses rejection and correction feedback from target verification to guide the promotion, retention, and eviction of task-relevant events.

Experiments on ProactiveVideoQA (Wang et al., 2025a), OVO-Bench (Niu et al., 2025), and StreamingBench (Lin et al., 2026) show that StreamBank largely preserves streaming video understanding performance even under substantial context compression, while achieving up to a $2.01\times$ decoding-throughput speedup.

Our main contributions are summarized as follows:

- We introduce **StreamBank**, a unified generation–memory inference framework for streaming video understanding. StreamBank verifies candidate blocks with the target model while reusing the resulting feedback to manage historical events, jointly reducing response-generation and context-processing costs.
- We introduce *Reuse-Augmented MTP Speculative Decoding*, which uses multiple MTP (Multi-Token Prediction) heads to predict future tokens at different positions in parallel, reducing autoregressive serial computation during candidate generation. When temporal continuity holds, it further reuses the previous target-verified response or its prefix to exploit response redundancy across adjacent streaming turns.
- We introduce *Speculation-Guided Event Memory*, which treats acceptance, rejection, and correction feedback from target verification as task-relevant signals for estimating the concrete contribution of historical events to the current prediction. It accordingly promotes, retains, and evicts events to preserve long-term information with greater predictive value under a fixed memory capacity.
- Experiments on ProactiveVideoQA, OVO-Bench, and StreamingBench show that StreamBank preserves streaming video understanding performance while compressing historical context and effectively improves decoding efficiency.

2 Related Work

Streaming Video Understanding. Streaming Video-LLMs extend video-language modeling from offline clips to continuously arriving visual streams, requiring causal perception, temporal context maintenance, and online interaction (Huang et al., 2025; Wang et al., 2024b; Xia et al., 2026). Existing studies mainly improve streaming inference through causal or streaming-oriented modeling, visual-token reduction, long-term context compression, and proactive response mechanisms that determine whether and when the model should respond (Lin et al., 2024; Bai et al., 2025). To control the growing inference cost of persistent streams, prior methods further rely on bounded recent windows, recurrent states, compressed representations, or retrieval-based memory to preserve useful historical information (Yao et al., 2025; Qian et al., 2025; Li et al., 2025). However, these approaches typically treat response generation and historical context management as separate problems: memory mechanisms decide what past information remains accessible, while the underlying decoder still generates responses independently in an autoregressive manner. In contrast, StreamBank couples generation and memory within a shared inference process, reusing target-verification signals produced during response generation to guide the retention of prediction-relevant historical events.

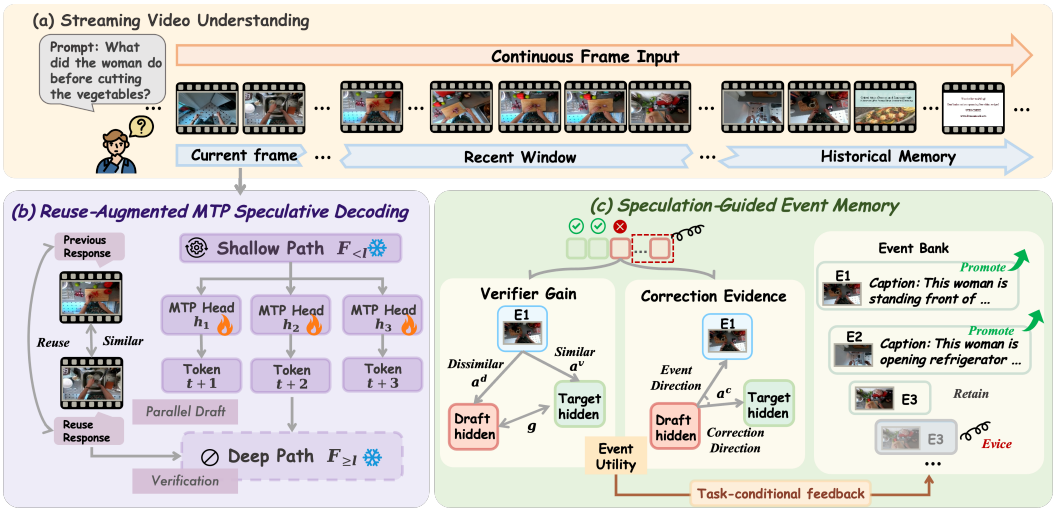

Figure 2: Overview of the proposed StreamBank. (a) Streaming video understanding maintains historical memory, a recent window, and the current frame under continuous visual input. (b) Reuse-Augmented MTP Speculative Decoding combines parallel multi-token drafting with cross-turn response reuse, followed by target-model verification through the frozen deep path. (c) Speculation-Guided Event Memory uses verification-derived gain and correction evidence to estimate event utility and dynamically promote, retain, or evict historical events.

Speculative Decoding and Multi-Token Prediction. Speculative decoding (SD) (Leviathan et al., 2023; Chen et al., 2023) accelerates autoregressive generation by proposing multiple future tokens with a lightweight mechanism and jointly verifying them with the target model. Existing methods reduce proposal overhead through separate draft models, auxiliary prediction heads (Cai et al., 2024), hidden-state prediction (Li et al., 2024a), dynamic candidate trees (Li et al., 2024b), and early-exit self-speculation (Elhoushi et al., 2024), while multi-token prediction (MTP) further enables several future positions to be predicted in parallel from a shared representation (Gloeckle et al., 2024). Recent multimodal extensions have adapted these ideas to vision-language models through modality-aware drafting (Lin et al., 2025), visual-context compression (Ji et al., 2025), and adaptive candidate construction (Wang et al., 2025c; Huo et al., 2025; Ji et al., 2026). Nevertheless, most existing methods are designed for a single generation episode, where verification feedback is used primarily to determine which proposed tokens can be accepted. They therefore do not explicitly exploit the strong cross-turn redundancy and persistent shared history unique to streaming video. StreamBank differs by extending speculative decoding across streaming turns: it combines parallel MTP with reuse of previous target-verified responses, and further repurposes verification feedback as an online signal for long-term event-memory management.

3 METHOD:STREAMBANK

3.1 FRAMEWORK OVERVIEW

StreamBank is a unified generation–memory inference framework for efficient streaming video understanding. It addresses two coupled sources of inference overhead: serial response generation and repeated processing of growing historical context. Rather than treating decoding acceleration and memory compression as independent components, StreamBank connects them through target-model verification.

As shown in Fig. 2, StreamBank consists of two coordinated components: *Reuse-Augmented MTP Speculative Decoding* and *Speculation-Guided Event Memory*. The former reduces sequential generation cost by combining parallel multi-token prediction with cross-turn reuse of previously target-verified responses. The latter maintains a fixed-capacity event memory and uses signals derived

from speculative verification to determine which historical events should be promoted, retained, or evicted. The target Video-LLM remains unchanged and serves as the final verifier for every generated token, while the lightweight proposal mechanism and memory-management policy operate around the frozen target model.

3.2 PRELIMINARIES

In streaming video understanding, a Video-LLM receives visual inputs sequentially and, at step t, has access only to the current and past observations, together with the instruction and interaction history. Its output is

$$O_t = f_\theta(X_{\le t}), \qquad O_t \in \{\varnothing, Y_t\}, \tag{1}$$

where $\varnothing$ denotes silence and Y_t denotes a textual response. StreamBank organizes the context as

$$X_t = M_t \oplus R_t \oplus x_t, \tag{2}$$

where x_t is the current observation, R_t is the recent multimodal window, and M_t is a fixed-capacity event memory. When a response is produced, the target Video-LLM generates it autoregressively:

$$p_\theta(Y_t \mid X_t) = \prod_{i=1}^{n} p_\theta(y_{t,i} \mid X_t, y_{t,<i}). \tag{3}$$

StreamBank preserves this target-model generation process while modifying two aspects of streaming inference: how candidate continuations are constructed during response generation, and how historical visual information is selected for long-term retention.

3.3 REUSE-AUGMENTED MTP SPECULATIVE DECODING

Conventional speculative decoding reduces serial target-model calls by verifying several candidate tokens in parallel, but its draft model commonly constructs those candidates autoregressively. This residual dependency is particularly costly in streaming video, where candidate generation is repeatedly invoked as new observations arrive. We remove the within-round dependency through Parallel MTP and complement it with a reuse mechanism that exploits redundancy across adjacent streaming turns.

Parallel MTP Drafting. We split the target decoder after layer l into a shallow path $F_{<l}$ and the remaining target path $F_{\ge l}$. For a committed prefix, the shallow path produces an early representation that a trainable adapter A_ϕ maps to

$$h_t = A_\phi(F_{<l}(X_t, Y_{t,<r}))\,, \tag{4}$$

where r is the next uncommitted response position. A set of K position-specific MTP heads transforms the same representation:

$$h_t^{(k)} = h_t + P_k(N_k(h_t))\,, \qquad k = 1, \dots, K, \tag{5}$$

where N_k and P_k are the normalization and residual projection of the k-th head. Each head independently predicts one future token,

$$d_k = \arg\max_v \, q_\phi^{(k)}\,(v \mid X_t, Y_{t,<r})\,, \qquad k = 1, \dots, K, \tag{6}$$

where d_k corresponds to the token at position $r + k - 1$.

Crucially, all heads condition on the same committed prefix rather than on predictions from preceding heads. StreamBank can thus form a candidate block in parallel instead of invoking the draft path once per candidate token. At inference, we take the greedy prediction of each head and retain the longest candidate prefix whose later-token confidence exceeds a threshold; the first head is always allowed to propose a token.

Verification-Gated Reuse. Streaming video often exhibits strong cross-turn redundancy: when the question is unchanged and adjacent visual states are similar, their target responses can also be highly consistent. StreamBank exploits this property by reusing the previous target-verified response as a candidate, avoiding a new drafting process when temporal continuity is strong.

Let v_{t-1} and v_t denote pooled visual keys of two adjacent turns, with $s_t = \cos(v_{t-1}, v_t)$. The reuse candidate is defined as

$$D_t^{\text{reuse}} = \begin{cases} Y_{t-1}, & s_t \geq \tau_{\text{high}}, \\ Y_{t-1}[:K_p], & \tau_{\text{low}} \leq s_t < \tau_{\text{high}}, \\ \varnothing, & s_t < \tau_{\text{low}}. \end{cases} \tag{7}$$

Reuse is disabled when the question changes. To ensure reliability, the candidate is retained only if its first token matches the target prefill prediction; otherwise, StreamBank falls back to Parallel MTP. The accepted reuse candidate is then verified by the target model in the same way as an MTP proposal.

Parallel Target Verification. Let $D = (d_1, \ldots, d_m)$ denote a candidate block produced by reuse or Parallel MTP. The target model evaluates all candidate positions and one additional position in a single forward pass, producing greedy tokens $(\hat{y}_1, \ldots, \hat{y}_{m+1})$. We define the accepted length as

$$a = \max\{j \in \{0, \ldots, m\} : d_i = \hat{y}_i \text{ for all } i \leq j\}. \tag{8}$$

StreamBank commits $(d_1, \ldots, d_a)$ and then the target token $\hat{y}_{a+1}$. The latter corrects the first mismatch when $a < m$ and becomes a bonus token when all m candidates are accepted. Generation repeats from the newly committed prefix until an end-of-sequence token is reached. Because no proposed token bypasses target verification, this rule preserves the target model's greedy continuation for a fixed context X_t. Importantly, when the target model rejects an MTP candidate, verification yields aligned draft and target representations at the mismatch position. Their discrepancy characterizes the information introduced by the target model to correct the candidate and provides task-conditioned evidence for updating event memory.

3.4 Speculation-Guided Event Memory

Retaining the complete streaming history makes the multimodal context grow with video duration, whereas keeping only a recent window may discard early evidence required by later responses. StreamBank therefore maintains a two-level context consisting of a recent multimodal-turn window $\mathcal{R}_t$ and a fixed-capacity event memory $\mathcal{M}_t$. Observations leaving the recent window are grouped into temporally coherent events m_j according to visual continuity. Each event stores its temporal span, a small set of representative frames, and an event representation e_j pooled from the corresponding target-model visual states. A newly formed event first remains temporary and becomes eligible for promotion only after receiving sufficient task-relevant support. Visual continuity thus determines how observations are organized, while target-verification feedback determines which events are semantically enriched and prioritized for long-term retention.

Verification-Derived Event Feedback. Speculative verification exposes positions where the lightweight proposal disagrees with the full target model, providing task-conditioned signals for identifying prediction-relevant historical events.

For the first rejected position i, let h_i^d and h_i^v denote the draft and position-aligned target representations, respectively. We use $h_i^v - h_i^d$ to characterize the correction direction and compute, for each visible event m_j,

$$a_{ij}^d = \cos(e_j, h_i^d), \qquad a_{ij}^v = \cos(e_j, h_i^v), \qquad a_{ij}^c = \cos(e_j, h_i^v - h_i^d). \tag{9}$$

The verifier gain is

$$g_{ij} = a_{ij}^v - a_{ij}^d, \tag{10}$$

which measures whether the event becomes more aligned with the target representation after correction. We then form two support signals:

$$b_{ij}^g = \mathbb{I}[g_{ij} > \delta_g], \qquad b_{ij}^c = \mathbb{I}[a_{ij}^c > \delta_c]. \tag{11}$$

To avoid bias toward longer responses, each event receives at most one gain count and one correction count per streaming turn, accumulated across turns as n_j^g and n_j^c. Repeated support indicates that an event remains relevant to target prediction or correction, allowing StreamBank to identify useful historical evidence beyond simple recency or visual similarity. Since all representations are already

produced during speculative verification, this feedback introduces almost no additional inference overhead.

Event Promotion and Eviction. Newly sealed events initially remain temporary. To avoid promoting an event because of an incidental match, we require support to accumulate across streaming turns. We normalize the verifier-gain and correction counts as

$$G_j = \min\left(1, \frac{n_j^g}{N_{\min}}\right), \qquad C_j = \min\left(1, \frac{n_j^c}{N_{\min}}\right), \tag{12}$$

and define the speculative-support score

$$S_j = \max(G_j, C_j), \tag{13}$$

where $N_{\min}$ is the number of supported streaming turns required for saturation. In addition, A_j indicates whether the event directly supported a non-silent response, and N_j denotes its visual novelty at formation. The promotion score is

$$P_j = \alpha_s \max(S_j, A_j) + \alpha_n N_j. \tag{14}$$

An event becomes eligible for promotion only after it is sealed, contains valid representative frames, obtains sufficient speculative or direct-response support, and satisfies $P_j \geq \tau_p$. StreamBank promotes at most one event per streaming turn. Promotion attaches a concise semantic description while retaining representative frames. When an event has already supported a response, that response can be reused as its description; otherwise, the target model generates a concise event caption.

Promotion and retention serve different purposes. Promotion determines whether an event has accumulated enough evidence to receive a semantic summary, whereas retention determines whether it should continue occupying the bounded event memory. We therefore define

$$R_j = \beta_s S_j + \beta_a A_j + \beta_n N_j + \beta_l L_j, \tag{15}$$

where $L_j \in \{0, 1\}$ indicates whether the event has been promoted. When the number of events exceeds capacity, StreamBank evicts the event with the smallest R_j, breaking ties in favor of older events.

Promotion does not make an event permanent: temporary and promoted events compete under the same retention criterion. Separating promotion from retention allows StreamBank to enrich events with stable task support while maintaining a fixed historical-context budget and prioritizing events that remain relevant under target verification.

4 Experiments

4.1 Datasets and Evaluation Benchmarks

Datasets. We adopt a two-stage training strategy. In the first stage, we use English ShareGPT conversations for multi-head draft alignment. In the second stage, we jointly train on a subset of MMDuet2 (Wang et al., 2026) and the full LLaVA-Video-178K (Zhang et al., 2025b) dataset to adapt the model to response-sparse streaming interactions while preserving general video-understanding capability. Further training details are provided in Appendix C.

Evaluation Benchmarks. We evaluate StreamBank on ProactiveVideoQA (Wang et al., 2025a), OVO-Bench (Niu et al., 2025), and StreamingBench (Lin et al., 2026). For ProactiveVideoQA, we report pAUC on the VAD, TV, Web, and Ego subsets, together with decoding speedup and average accepted length. For OVO-Bench and StreamingBench, we follow the official evaluation protocols and report the corresponding task metrics. We additionally report results on several general video-understanding benchmarks in Appendix D.

4.2 Main Results

Results on OVO-Bench and StreamingBench. Table 1 presents the real-time evaluation results on OVO-Bench and StreamingBench. On OVO-Bench, StreamBank achieves an average score of 62.7, improving the MMDuet2 target model by 0.7 points and outperforming all other open-source

Table 1: Results on the real-time subtasks of OVO-Bench and StreamingBench.

Method	Size	Frames	OVO-Bench Real-Time							StreamingBench Real-Time										
			OCR	ACR	ATR	STU	FPD	OJR	Avg.	OP	CR	CS	ATP	EU	TR	PR	SU	ACP	CT	Avg.
Human																				
Human	–	–	94.0	92.6	94.8	92.7	91.1	94.0	93.2	89.5	92.0	93.6	91.5	95.7	92.5	88.0	88.8	89.7	91.3	91.5
Proprietary Models																				
Gemini 1.5 Pro	–	1 fps	85.9	67.0	79.3	58.4	63.4	62.0	69.3	79.0	80.5	83.5	79.7	80.0	84.7	77.8	64.2	72.0	48.7	75.7
GPT-4o	–	64	69.8	64.2	71.6	51.1	70.3	59.8	64.5	77.1	80.5	83.9	76.5	70.2	83.8	66.7	62.2	69.1	49.2	73.3
Claude 3.5 Sonnet	–	20	69.8	64.2	71.6	51.1	70.3	59.8	64.5	73.3	80.5	84.1	82.0	75.4	79.5	61.1	61.8	69.3	49.1	72.4
Open-source Offline MLLMs																				
LongVA	7B	128	–	–	–	–	–	–	–	70.0	63.3	61.2	70.9	62.7	59.5	61.1	53.7	54.7	34.7	60.0
LongVU	7B	1 fps	55.7	49.5	59.5	48.3	68.3	63.0	57.4	–	–	–	–	–	–	–	–	–	–	–
InternVL2	8B	64/16	67.1	60.6	63.8	46.1	68.3	56.5	60.4	68.1	60.9	69.4	77.1	67.7	62.9	59.3	53.3	55.0	56.5	63.7
Qwen2.5-VL	7B	64	–	–	–	–	–	–	–	78.3	80.5	79.8	82.4	75.5	80.4	74.1	62.6	67.6	51.1	73.9
Open-source Online MLLMs																				
VideoLLM-Online	8B	2 fps	8.1	23.9	12.1	14.0	45.5	21.2	20.8	39.1	40.1	34.5	31.1	46.0	32.4	31.5	34.2	42.5	27.9	36.0
Dispider	7B	1 fps	57.7	49.5	62.1	44.9	61.4	51.6	54.6	74.9	75.5	74.1	73.1	74.4	59.9	76.1	62.9	62.2	45.8	67.6
Flash-VStream	7B	1 fps	24.2	29.4	28.5	33.7	25.7	28.8	28.4	25.9	43.6	24.9	23.9	27.3	13.1	18.5	25.2	23.9	48.7	23.2
TimeChat-Online	7B	1 fps	74.5	48.6	68.1	48.3	69.3	**59.8**	61.4	80.8	79.7	80.8	83.3	74.8	78.8	78.7	64.2	68.8	**58.0**	75.3
ReKV	7B	0.5 fps	–	–	–	–	–	–	–	74.4	78.9	78.6	77.1	68.3	67.9	67.6	62.6	64.3	44.6	69.1
LiveVLM	7B	0.5 fps	–	–	–	–	–	–	–	81.5	78.1	83.3	79.1	69.6	74.1	75.0	**69.1**	67.7	40.4	72.9
StreamAgent	7B	1 fps	71.2	53.2	63.6	**53.9**	67.3	58.7	61.3	79.6	78.3	79.3	75.9	74.7	76.9	**82.9**	66.3	**73.7**	55.4	74.3
QueryStream	7B	1 fps	**74.5**	47.7	**70.7**	46.6	**71.3**	57.6	61.4	82.1	**83.6**	78.2	82.7	75.5	80.1	79.6	63.0	67.9	42.6	74.0
MMDuet2†	3B	1 fps	71.1	61.5	69.0	46.1	65.4	59.2	62.0	81.2	78.1	86.4	83.0	81.3	78.8	75.0	67.5	67.4	40.9	74.9
StreamBank	3B	1 fps	69.1	**61.5**	69.8	46.1	70.3	59.2	**62.7**	**82.3**	78.9	**86.8**	**83.7**	**81.4**	**80.4**	76.9	68.3	68.6	42.0	**75.9**

online MLLMs. StreamBank matches or improves MMDuet2 on five of the six task categories, with the largest gain of 4.9 points on FPD. On StreamingBench, StreamBank improves all ten task-specific scores over MMDuet2 and raises the average score from 74.9 to 75.9. Despite using only a 3B target model, StreamBank also surpasses every competing method, including the proprietary Gemini 1.5 Pro at 75.7. These results show that the bounded event memory retains prediction-relevant historical information and that the unified generation–memory framework preserves, and often improves, streaming video understanding performance.

Table 2: Experimental results on ProactiveVideoQA. We report pAUC / Speedup to jointly characterize response quality and decoding efficiency. Speedup denotes the ratio of decoding throughput, measured in tokens per second, to that of the base model. AAL denotes the average accepted length. † denotes results reproduced using Qwen3-32B during evaluation.

Method	WEB	EGO	TV	VAD	Overall	AAL
VideoLLM-Online	25.9	25.0	18.3	25.0	23.6	–
MMDuet2†	36.0	**28.1**	30.6	25.5	30.1	–
StreamBank$_{w/o\ \text{reuse}}$	36.1 / 1.32×	27.7 / 1.84×	32.7 / 1.56×	26.9 / 1.84×	30.8 / 1.69×	2.49
StreamBank	**36.2 / 1.33**×	27.7 / **2.42**×	**33.0 / 1.69**×	**27.1 / 2.26**×	**31.0 / 2.01**×	**2.61**

Results on ProactiveVideoQA. ProactiveVideoQA evaluates both response timing and content generation, directly testing whether faster decoding compromises proactive interaction. As shown in Table 2, StreamBank outperforms the reproduced MMDuet2 baseline on three subsets, with larger gains on TV and VAD, indicating that bounded event memory preserves response-relevant temporal evidence while filtering redundant history. Meanwhile, StreamBank achieves a $2.01\times$ decoding-throughput speedup with an AAL of 2.61, showing that most parallel proposals pass target verification. We also report StreamBank$_{w/o\ \text{reuse}}$. Without Verification-Gated Reuse, parallel MTP still provides substantial acceleration but is less efficient than the full model, confirming that reuse further reduces drafting overhead by exploiting temporal continuity. The quality differences are small, while the principal observed benefit of reuse is higher decoding throughput.

4.3 Ablation Studies

Effect of Event Retention Policy. We evaluate the historical-event retention ability of our Speculation-Guided Event Memory on the Episodic Memory (EPM) subtask of OVO-Bench. EPM requires the model to recall a concrete event, object state, or object location observed earlier in the

video stream, directly testing whether relevant evidence remains accessible after leaving the recent window. We compare StreamBank against three retention policies: (i) *FIFO*, which retains the most recently sealed events up to the bank capacity; (ii) *visual similarity*, which retains the historical events most similar to the current observation up to the bank capacity; and (iii) *full history*, the MMDuet2-style baseline that retains the complete visual history without event selection.

Table 3: Comparison of event-retention policies on the OVO-Bench EPM subtask.

Retention policy	EPM accuracy (%)
FIFO	51.18
Visual similarity	49.83
Full history	48.82
StreamBank	**52.53**

As shown in Table 3, StreamBank achieves 52.53% EPM accuracy. Under the same bounded-memory budget, it outperforms FIFO and visual similarity by 1.35 and 2.70 points, respectively; despite retaining less visual context, it also exceeds full history by 3.71 points. These results show that StreamBank preserves episodic evidence more effectively than retention policies based solely on recency or current-frame similarity. Its improvement over full history further indicates that StreamBank not only compresses the visual context, but also filters redundant information while retaining the critical evidence required for backward-tracing questions.

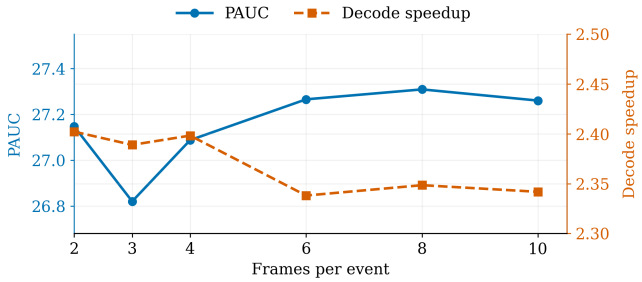

Figure 3: Effect of the representative-frame budget on VAD quality and decoding speedup.

Effect of Representative Frame Budget. We vary the number of representative frames retained for each event on the VAD subset of ProactiveVideoQA. As shown in Figure 3, retaining more representative frames preserves richer event information and generally improves pAUC, but also increases memory consumption and the amount of visual context processed during inference. Therefore, we retain two representative frames for each event by default, achieving a pAUC of 27.14 and a $2.40\times$ decoding speedup, which provides a favorable balance between performance and efficiency.

5 Conclusion

We presented StreamBank, a unified generation–memory inference framework for efficient streaming video understanding. Reuse-Augmented MTP Speculative Decoding combines parallel multi-token prediction with cross-turn response reuse to reduce sequential generation overhead, while the unchanged target model verifies every proposed token. Speculation-Guided Event Memory reuses verifier-gain and correction signals to guide the promotion, retention, and eviction of historical events under a fixed memory capacity. Experiments on ProactiveVideoQA, OVO-Bench, and StreamingBench show that StreamBank preserves streaming understanding performance while achieving up to a $2.01\times$ decoding-throughput speedup. Results on long-video benchmarks further confirm that verification-guided memory selection retains prediction-relevant evidence under bounded context. Overall, StreamBank unifies generation acceleration and memory management through target-model verification.

REPRODUCIBILITY STATEMENT

We provide the implementation and experimental details required to reproduce StreamBank. Section 3 describes the complete framework, including Reuse-Augmented MTP Speculative Decoding, Verification-Gated Reuse, parallel target-model verification, and Speculation-Guided Event Memory. The complete decoding procedure and its greedy-correctness argument are provided in Appendix A. Section 4 specifies the training datasets, evaluation benchmarks, and evaluation protocols, including ProactiveVideoQA, OVO-Bench, and StreamingBench. Additional training configurations, implementation details, and hyperparameters are provided in Appendix B and Appendix C. We also report the settings used for the reproduced MMDuet2 baseline, the no-reuse variant, event-retention baselines, and other ablation studies to ensure consistent evaluation and comparison. During the review process, we provide an anonymous code repository containing the implementation, configuration files, and evaluation scripts required to reproduce the reported results.

AI USE STATEMENT

Generative AI tools were used solely to assist with language editing and polishing the writing of this manuscript. The authors reviewed and revised all AI-assisted text and take full responsibility for the final content of the paper. No AI-generated content was used to produce the scientific results, analyses, or conclusions.

REFERENCES

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025. URL https://arxiv.org/abs/2502.13923.

Tianle Cai, Yuhong Li, Zhengyang Geng, Hongwu Peng, Jason D. Lee, Deming Chen, and Tri Dao. Medusa: Simple LLM inference acceleration framework with multiple decoding heads. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 5209–5235. PMLR, 2024.

Charlie Chen, Sebastian Borgeaud, Geoffrey Irving, Jean-Baptiste Lespiau, Laurent Sifre, and John Jumper. Accelerating large language model decoding with speculative sampling. *arXiv preprint arXiv:2302.01318*, 2023. URL https://arxiv.org/abs/2302.01318.

Joya Chen, Zhaoyang Lv, Shiwei Wu, Kevin Qinghong Lin, Chenan Song, Difei Gao, Jia-Wei Liu, Ziteng Gao, Dongxing Mao, and Mike Zheng Shou. 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. IEEE, 2024. doi: 10.1109/CVPR52733.2024.01742.

Joya Chen, Ziyun Zeng, Yiqi Lin, Wei Li, Zejun Ma, and Mike Zheng Shou. LiveCC: Learning video LLM with streaming speech transcription at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 29083–29095, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Chen_LiveCC_Learning_Video_LLM_with_Streaming_Speech_Transcription_at_Scale_CVPR_2025_paper.html.

Shangzhe Di, Zhelun Yu, Guanghao Zhang, Haoyuan Li, Hao Cheng, Bolin Li, Wanggui He, Fangxun Shu, Hao Jiang, et al. Streaming video question-answering with in-context video kv-cache retrieval. In *ICLR*, 2025.

Xin Ding, Hao Wu, Yifan Yang, Shiqi Jiang, Qianxi Zhang, Donglin Bai, Zhibo Chen, and Ting Cao. StreamMind: Unlocking full frame rate streaming video dialogue through event-gated cognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13448–13459. IEEE, 2025.

Mostafa Elhoushi, Akshat Shrivastava, Diana Liskovich, Basil Hosmer, Bram Wasti, Liangzhen Lai, Anas Mahmoud, Bilge Acun, Saurabh Agarwal, Ahmed Roman, Ahmed A. Aly, Beidi Chen, and Carole-Jean Wu. LayerSkip: Enabling early exit inference and self-speculative decoding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12622–12642, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.681.

Fabian Gloeckle, Badr Youbi Idrissi, Baptiste Roziere, David Lopez-Paz, and Gabriel Synnaeve. Better & faster large language models via multi-token prediction. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 15706–15734. PMLR, 2024. URL https://proceedings.mlr.press/v235/gloeckle24a.html.

Kristen Grauman, Andrew Westbury, Lorenzo Torresani, et al. Ego-Exo4D: Understanding skilled human activity from first- and third-person perspectives. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19383–19400, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Grauman_Ego-Exo4D_Understanding_Skilled_Human_Activity_from_First-_and_Third-Person_Perspectives_CVPR_2024_paper.html.

Yifei Huang, Guo Chen, Jilan Xu, Mingfang Zhang, Lijin Yang, Baoqi Pei, Hongjie Zhang, Lu Dong, Yali Wang, Limin Wang, and Yu Qiao. EgoExoLearn: A dataset for bridging asynchronous ego- and exo-centric view of procedural activities in real world. In *Proceedings*

of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 22072–22086, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Huang_EgoExoLearn_A_Dataset_for_Bridging_Asynchronous_Ego-_and_Exo-centric_View_CVPR_2024_paper.html.

Zhenpeng Huang, Xinhao Li, Jiaqi Li, Jing Wang, Xiangyu Zeng, Cheng Liang, Tao Wu, Xi Chen, Liang Li, and Limin Wang. Online video understanding: OVBench and VideoChat-Online. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3328–3338. IEEE, 2025.

Mingxiao Huo, Jiayi Zhang, Hewei Wang, Jinfeng Xu, Zheyu Chen, Huilin Tai, and Yijun Chen. Spec-LLaVA: Accelerating vision-language models with dynamic tree-based speculative decoding. *arXiv preprint arXiv:2509.11961*, 2025. doi: 10.48550/arXiv.2509.11961. URL https://arxiv.org/abs/2509.11961.

Yicheng Ji, Jun Zhang, Heming Xia, Jinpeng Chen, Lidan Shou, Gang Chen, and Huan Li. Specvlm: Enhancing speculative decoding of video llms via verifier-guided token pruning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 7216–7230, 2025.

Yicheng Ji, Jun Zhang, Jinpeng Chen, Cong Wang, Lidan Shou, Gang Chen, and Huan Li. See the forest for the trees: Loosely speculative decoding via visual-semantic guidance for efficient inference of video LLMs. *arXiv preprint arXiv:2604.05650*, 2026. doi: 10.48550/arXiv.2604.05650. URL https://arxiv.org/abs/2604.05650.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19274–19286. PMLR, 2023. URL https://proceedings.mlr.press/v202/leviathan23a.html.

Wei Li, Bing Hu, Rui Shao, Leyang Shen, and Liqiang Nie. LION-FS: Fast & slow video-language thinker as online video assistant. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3240–3251. IEEE, 2025.

Yuhui Li, Fangyun Wei, Chao Zhang, and Hongyang Zhang. EAGLE: Speculative sampling requires rethinking feature uncertainty. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 28935–28948. PMLR, 2024a.

Yuhui Li, Fangyun Wei, Chao Zhang, and Hongyang Zhang. EAGLE-2: Faster inference of language models with dynamic draft trees. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 7421–7432, Miami, FL, USA, 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.422.

Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. 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. doi: 10.18653/v1/2024.emnlp-main.342. URL https://aclanthology.org/2024.emnlp-main.342/.

Junming Lin, Zheng Fang, Chi Chen, Haoxuan Cheng, Zihao Wan, Fuwen Luo, Ziyue Wang, Peng Li, Yang Liu, and Maosong Sun. StreamingBench: Assessing the gap for MLLMs to achieve streaming video understanding. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 12147–12151, 2026. doi: 10.1109/ICASSP55912.2026.11463959. URL https://doi.org/10.1109/ICASSP55912.2026.11463959.

Luxi Lin, Zhihang Lin, Zhanpeng Zeng, and Rongrong Ji. Speculative decoding reimagined for multimodal large language models. *arXiv preprint arXiv:2505.14260*, 2025. doi: 10.48550/arXiv.2505.14260. URL https://arxiv.org/abs/2505.14260.

Fangcheng Liu, Yehui Tang, Zhenhua Liu, Yunsheng Ni, Kai Han, and Yunhe Wang. Kangaroo: Lossless self-speculative decoding via double early exiting, 2024. URL https://arxiv.org/abs/2404.18911.

Zhenyu Ning, Guangda Liu, Qihao Jin, Chengwei Li, Wenchao Ding, Minyi Guo, and Jieru Zhao. LiveVLM: Efficient online video understanding via streaming-oriented KV cache and retrieval. In *Proceedings of the 63rd ACM/IEEE Design Automation Conference*, pp. 1–6, Long Beach, CA, USA, 2026. Association for Computing Machinery. doi: 10.1145/3770743.3804012.

Junbo Niu, Yifei Li, Ziyang Miao, Chunjiang Ge, Yuanhang Zhou, Qihao He, Xiaoyi Dong, Haodong Duan, Shuangrui Ding, Rui Qian, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, and Jiaqi Wang. OVO-Bench: How far is your video-LLMs from real-world online video understanding? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18902–18913, 2025. doi: 10.1109/CVPR52734.2025.01761. URL https://openaccess.thecvf.com/content/CVPR2025/html/Niu_OVO-Bench_How_Far_is_Your_Video-LLMs_from_Real-World_Online_Video_CVPR_2025_paper.html.

Rui Qian, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Shuangrui Ding, Dahua Lin, and Jiaqi Wang. Streaming long video understanding with large language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 119336–119360. Curran Associates, Inc., 2024.

Rui Qian, Shuangrui Ding, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Dahua Lin, and Jiaqi Wang. Dispider: Enabling video LLMs with active real-time interaction via disentangled perception, decision, and reaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24045–24055. IEEE, 2025. doi: 10.1109/CVPR52734.2025.02239.

Yujiao Shen, Shulin Tian, Jingkang Yang, and Ziwei Liu. A simple baseline for streaming video understanding, 2026. URL https://arxiv.org/abs/2604.02317.

Weihan Wang, Zehai He, Wenyi Hong, Yean Cheng, Xiaohan Zhang, Ji Qi, Shiyu Huang, Bin Xu, Yuxiao Dong, Ming Ding, and Jie Tang. Lvbench: An extreme long video understanding benchmark, 2024a.

Yueqian Wang, Xiaojun Meng, Yuxuan Wang, Jianxin Liang, Jiansheng Wei, Huishuai Zhang, and Dongyan Zhao. Videollm knows when to speak: Enhancing time-sensitive video comprehension with video-text duet interaction format. *arXiv preprint arXiv:2411.17991*, 1(3):5, 2024b.

Yueqian Wang, Xiaojun Meng, Yifan Wang, Huishuai Zhang, and Dongyan Zhao. ProactiveVideoQA: A comprehensive benchmark evaluating proactive interactions in video large language models. *arXiv preprint arXiv:2507.09313*, 2025a. URL https://arxiv.org/abs/2507.09313.

Yueqian Wang, Songxiang Liu, Disong Wang, Nuo Xu, Guanglu Wan, Huishuai Zhang, and Dongyan Zhao. MMDuet2: Enhancing proactive interaction of video MLLMs with multi-turn reinforcement learning. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/281970ef377580ab9ec3c7d3781110d4-Abstract-Conference.html.

Yuxuan Wang, Yiqi Song, Cihang Xie, Yang Liu, and Zilong Zheng. Videollamb: Long streaming video understanding with recurrent memory bridges. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 24170–24181, October 2025b.

Zihua Wang, Ruibo Li, Haozhe Du, Joey Tianyi Zhou, Yu Zhang, and Xu Yang. FLASH: Latent-aware semi-autoregressive speculative decoding for multimodal tasks. *arXiv preprint arXiv:2505.12728*, 2025c. doi: 10.48550/arXiv.2505.12728. URL https://arxiv.org/abs/2505.12728.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. Longvideobench: A benchmark for long-context interleaved video-language understanding, 2024. URL https://arxiv.org/abs/2407.15754.

Jiaer Xia, Peixian Chen, Mengdan Zhang, Xing Sun, and Kaiyang Zhou. Streaming video instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 31219–31229, June 2026.

Zhenyu Yang, Kairui Zhang, Yuhang Hu, Bing Wang, Shengsheng Qian, Bin Wen, Fan Yang, Tingting Gao, Weiming Dong, and Changsheng Xu. Livestar: Live streaming assistant for real-world online video understanding. *Advances in Neural Information Processing Systems*, 38:31266–31304, 2026.

Linli Yao, Yicheng Li, Yuancheng Wei, Lei Li, Shuhuai Ren, Yuanxin Liu, Kun Ouyang, Lean Wang, Shicheng Li, Sida Li, Lingpeng Kong, Qi Liu, Yuanxing Zhang, and Xu Sun. TimeChat-Online: 80% visual tokens are naturally redundant in streaming videos. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 10807–10816, Dublin, Ireland, 2025. Association for Computing Machinery. doi: 10.1145/3746027.3754839.

Xiangyu Zeng, Kefan Qiu, Qingyu Zhang, Xinhao Li, Jing Wang, Jiaxin Li, Ziang Yan, Kun Tian, Meng Tian, Xinhai Zhao, Yi Wang, and Limin Wang. Streamforest: Efficient online video understanding with persistent event memory, 2025. URL https://arxiv.org/abs/2509.24871.

Haoji Zhang, Yiqin Wang, Yansong Tang, Yong Liu, Jiashi Feng, and Xiaojie Jin. Flash-VStream: Efficient real-time understanding for long video streams. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 21059–21069. IEEE, 2025a.

Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. LLaVA-Video: Video instruction tuning with synthetic data. *Transactions on Machine Learning Research*, 2025b. ISSN 2835-8856. URL https://openreview.net/forum?id=EElFGvt39K.

Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Shitao Xiao, Xi Yang, Yongping Xiong, Bo Zhang, Tiejun Huang, and Zheng Liu. Mlvu: A comprehensive benchmark for multi-task long video understanding. *arXiv preprint arXiv:2406.04264*, 2024.

APPENDIX

A PARALLEL MTP DECODING PROCEDURE AND GREEDY CORRECTNESS

Algorithm 1 StreamBank decoding with Parallel MTP and verified temporal reuse

Input: current input x_t, recent window $\mathcal{R}_t$, event bank M_t, frozen target p_θ, MTP heads $q_\phi^{(1:K)}$
Fix $X_t \leftarrow \mathcal{M}_t \oplus \mathcal{R}_t \oplus x_t$
Target prefill: $y_1 \leftarrow \arg\max_z p_\theta(z \mid X_t)$; set $u \leftarrow (y_1)$
$D_{\text{reuse}} \leftarrow \text{TemporalSuffix}(t, y_1)$ if the question and visual-continuity gates pass; otherwise $\varnothing$
while u has no end token and the output limit is not reached **do**
  $D \leftarrow D_{\text{reuse}}$; set $D_{\text{reuse}} \leftarrow \varnothing$ (reuse is tried at most once)
  if $D = \varnothing$ **then**
    Compute one shared shallow representation from (X_t, u)
    Predict $d_k \leftarrow \arg\max_z q_\phi^{(k)}(z \mid X_t, u)$ for all $k = 1, \ldots, K$ in parallel
    $D \leftarrow (d_1, \ldots, d_m)$, the longest confidence-qualified prefix; $m \geq 1$
  end if
  Let $D = (d_1, \ldots, d_m)$; verify $v_i \leftarrow \arg\max_z p_\theta(z \mid X_t, u, d_{<i})$ for $i = 1, \ldots, m+1$ in one causal pass
  $a \leftarrow \max\{j \in \{0, \ldots, m\} : d_i = v_i \text{ for all } i \leq j\}$
  Commit $d_{1:a}$, then v_{a+1}, stopping at the first end token
  Discard uncommitted candidate tokens and cache entries; append committed tokens to u
  if $a < m$ and D came from MTP **then**
    Record aligned draft–verifier feedback for visible events
  end if
end while
return u; update event-memory utility for subsequent turns

Algorithm 1 gives the complete decoding procedure, whose correctness follows from a prefix invariant. At streaming turn t, StreamBank updates the recent window and event bank and fixes the context for the entire response as

$$X_t = \mathcal{M}_t \oplus \mathcal{R}_t \oplus x_t. \tag{16}$$

The target model first obtains the initial greedy token by prefill. Given the committed prefix $u = Y_{t,<r}$, the first speculative round may reuse the previous target-verified response when the question and visual-continuity gates pass. Reuse is enabled only if its first token agrees with the current target-prefill prediction, after which the remaining suffix is used as the candidate block; this reuse is attempted at most once.

Otherwise, one shared shallow representation feeds K MTP heads, which predict future tokens in parallel:

$$d_k = \arg\max_v q_\phi^{(k)}(v \mid X_t, u), \qquad k = 1, \ldots, K. \tag{17}$$

The longest confidence-qualified prefix forms

$$D = (d_1, \ldots, d_m). \tag{18}$$

For either proposal source, a single causally masked target-verification pass computes

$$v_i = \arg\max_z p_\theta(z \mid X_t, u, d_{<i}), \qquad i = 1, \ldots, m+1. \tag{19}$$

The accepted length is

$$a = \max\{j \in \{0, \ldots, m\} : d_i = v_i,\ \forall i \leq j\}. \tag{20}$$

The decoder commits the accepted prefix $d_{1:a}$ followed by v_{a+1}, which serves as the first-mismatch correction when $a < m$ and as a bonus token when $a = m$. Uncommitted candidates and their cache entries are discarded, and decoding continues until an end token is produced or the output limit is reached.

To establish greedy equivalence, we assume deterministic greedy tie-breaking, causal block verification equivalent to token-by-token target evaluation, and consistent positions and caches. Let the greedy sequence of the unchanged target model under the fixed context X_t be

$$y_i^* = \arg\max_z p_\theta(z \mid X_t, y_{<i}^*). \tag{21}$$

The empty prefix and the target-prefill token trivially agree with this sequence. Suppose the committed prefix u agrees with the target greedy prefix at the beginning of a speculative round. At every accepted position $i \le a$, all preceding candidates have already been verified, so $u \parallel d_{<i}$ is also a target greedy prefix and

$$d_i = v_i = y_{r+i-1}^*. \tag{22}$$

After the accepted candidates, the next target token satisfies

$$v_{a+1} = \arg\max_z p_\theta(z \mid X_t, u, d_{1:a}) = y_{r+a}^*. \tag{23}$$

Thus, every committed token preserves the prefix invariant, regardless of whether the proposal comes from Parallel MTP or temporal reuse, and each round advances by at least one token. The final response therefore matches the unchanged target model's greedy response under the *same* X_t, or its prefix when the output limit is reached. Changes to the event bank alter X_t and are therefore outside this equivalence claim. Rejected MTP candidates additionally provide aligned draft and verifier states for updating event feedback in subsequent turns without affecting already committed tokens.

B Experiment Settings

B.1 Datasets and Evaluation Benchmarks

Datasets. We train StreamBank using a two-stage curriculum. In the first stage, *Multi-Head Base-Model Draft Alignment*, we use English ShareGPT conversations to initialize the multi-token prediction heads and align their predictions with the frozen target model. The conversations are tokenized and segmented into sequences of at most 2,048 tokens. Since this stage contains only textual inputs, it introduces no additional visual-processing cost and mainly establishes the basic multi-head drafting capability.

In the second stage, *Streaming-Aware Multi-Head Adaptation*, we further adapt the prediction heads using an 8,855-sample subset derived from MMDuet2 (Wang et al., 2026) together with the complete LLaVA-Video-178K dataset (Zhang et al., 2025b). The MMDuet2 subset contains approximately 7.97k training samples and 883 validation samples collected from Live-WhisperX (Chen et al., 2025), Ego-Exo4D (Grauman et al., 2024), and EgoExoLearn (Huang et al., 2024), covering both egocentric and exocentric activities. These streaming trajectories expose the model to response-sparse interactions and temporally evolving observations, while LLaVA-Video-178K provides diverse video-instruction examples for preserving general video-understanding capability. Further implementation and optimization details are provided in Appendix C.

Streaming Evaluation Benchmarks. We evaluate the streaming capability of StreamBank on ProactiveVideoQA (Wang et al., 2025a), OVO-Bench (Niu et al., 2025), and StreamingBench (Lin et al., 2026), following their official evaluation protocols.

ProactiveVideoQA evaluates proactive interaction in continuous video streams, where the model must determine not only what to answer but also when a response should be produced. We report the official pAUC scores on the VAD, TV, Web, and Ego subsets. Since ProactiveVideoQA involves actual response generation, we additionally report decoding throughput speedup and average accepted length (AAL) to evaluate the efficiency of speculative decoding.

OVO-Bench evaluates online video understanding under continuously arriving visual observations and contains multiple real-time tasks covering different forms of temporal and event-level reasoning. StreamingBench provides a complementary evaluation of real-time streaming understanding across a broad range of online video tasks. For both benchmarks, we follow their official input construction and scoring procedures and report the corresponding task-wise and average scores. Because these benchmarks are primarily evaluated through multiple-choice prediction rather than free-form autoregressive generation, decoding speedup is not reported on them.

General and Long-Video Evaluation Benchmarks. To examine whether the proposed generation and memory mechanisms preserve general video-understanding ability beyond streaming-specific settings, we further evaluate StreamBank on LongVideoBench (Wu et al., 2024), MLVU (Zhou et al., 2024), and LVBench (Wang et al., 2024a). These benchmarks emphasize long-duration video understanding and require the model to integrate information distributed over extended temporal contexts.

For LongVideoBench, we follow the official evaluation protocol and report the corresponding multiple-choice accuracy. For MLVU, we report the official task-level results and the overall average across its holistic, single-detail, and multi-detail evaluation categories. For LVBench, we report the official per-category scores together with the overall score. These additional evaluations are used to examine whether bounded event memory and speculative generation retain long-range video evidence without degrading general video-understanding performance.

Unless otherwise specified, all comparisons use the same target model and follow the benchmark-specific input, prompting, and scoring protocols.

C Parallel MTP Training

We train Parallel MTP with a two-stage draft-to-stream curriculum extended to K parallel prediction heads. The target Video-LLM remains frozen throughout both stages. The first stage aligns every MTP head with the target model's greedy behavior, and the second adapts the parallel proposal mechanism to response-sparse streaming trajectories.

Stage 1: Multi-Head Base-Model Draft Alignment. For an input context X_t, we first obtain the greedy response $\bar{Y}_t = (\bar{y}_{t,1}, \ldots, \bar{y}_{t,n})$ from the frozen target model. At response position i, the k-th MTP head is trained to predict the target-model token $\bar{y}_{t,i+k-1}$ from the shared representation conditioned on $\bar{Y}_{t,<i}$. Let $\mathcal{D}_k$ contain the positions for which the k-step target is valid. We optimize

$$\mathcal{L}_{\text{align}} = -\frac{1}{K}\sum_{k=1}^{K}\frac{1}{|\mathcal{D}_k|}\sum_{i\in\mathcal{D}_k}\log q_\phi^{(k)}\left(\bar{y}_{t,i+k-1} \mid X_t, \bar{Y}_{t,<i}\right). \tag{24}$$

Unlike single-token draft alignment, the objective aligns the entire parallel candidate block with the corresponding future positions of the target model. It therefore directly trains the quantity that controls speculative efficiency: the length of the prefix accepted by target verification.

Stage 2: Streaming-Aware Multi-Head Adaptation. Base-model alignment alone does not expose the proposal mechanism to the response sparsity and temporal transitions of online video. We therefore continue training on streaming interaction trajectories. Given a reference response Y_t, each MTP head predicts its corresponding future token from the same response prefix:

$$\mathcal{L}_{\text{stream}} = -\frac{1}{K}\sum_{k=1}^{K}\frac{1}{|\mathcal{R}_k|}\sum_{i\in\mathcal{R}_k}\log q_\phi^{(k)}(y_{t,i+k-1} \mid X_t, Y_{t,<i}), \tag{25}$$

where $\mathcal{R}_k$ denotes the supervised response positions with a valid target for head k. This stage adapts all prediction heads jointly while preserving their parallel conditioning structure. At deployment, the trained adapter, MTP heads, and draft-only vocabulary head are used exclusively for candidate generation; autoregressive generation and parallel verification continue to use the unchanged target model. The event-memory policy requires no additional training and obtains its task evidence online from target verification.

D More Results

D.1 Implementation Details

We use our reproduced MMDuet2 model based on Qwen2.5-VL-3B as the target model. During training, the target model and its original language-modeling head remain frozen. We only optimize a three-layer adapter, three position-specific MTP heads, and a separate draft language-modeling head. The adapter branches from decoder layer $l = 4$, and we use $K = 3$ MTP heads, allowing

Table 4: Experimental results on LongVideoBench.

Method	E2O	E3E	O2E	O3O	S2A	S2E	S2O	SAA	SOS	SSS	T2A	T2E	T2O	T3E	T3O	TAA	TOS	Avg.
GPT-4o	67.3	62.6	72.8	61.3	70.9	76.8	69.8	62.6	75.6	44.3	77.2	67.2	65.3	64.0	66.4	66.4	62.1	66.7
Gemini-1.5-Pro	68.7	61.9	73.3	58.6	76.2	74.6	58.3	59.0	69.0	55.2	76.7	66.2	63.6	58.9	60.5	62.5	53.3	64.4
Gemini-1.5-Flash	64.5	56.5	72.5	61.0	68.0	68.5	64.7	56.2	67.3	43.1	76.7	63.6	68.0	57.5	55.0	60.7	55.3	62.4
LLaVA-1.5-13B	44.8	45.2	49.0	40.9	40.4	54.9	42.6	44.2	53.3	29.9	43.0	51.1	43.1	38.7	35.6	46.2	30.0	43.1
LLaVA-1.5-7B	39.6	40.7	44.9	43.9	38.8	53.3	35.0	42.9	47.3	26.2	43.3	44.1	39.9	37.2	34.7	45.1	30.3	40.4
VideoChat2	44.5	43.6	53.5	46.6	38.5	53.6	40.8	40.6	42.0	10.6	47.7	46.8	43.1	38.4	36.3	43.6	27.4	41.2
VideoLLaVA	40.9	46.5	44.9	38.7	30.6	49.5	29.6	42.9	40.0	24.3	40.6	43.5	33.8	35.1	30.5	39.5	23.8	37.6
MMDuet2†	63.1	51.1	56.3	51.5	55.7	60.2	58.3	45.8	58.0	32.0	49.4	50.8	56.6	43.8	47.3	45.1	37.0	50.6
StreamBank	**66.2**	**58.5**	**62.1**	**54.5**	**60.2**	**69.9**	**61.1**	**47.2**	**67.9**	**39.2**	**55.7**	**60.0**	**63.2**	**54.8**	**52.7**	**47.6**	**39.7**	**56.5**

up to three draft tokens to be proposed in parallel. The MTP confidence threshold is set to 0.6. For streaming context management, the recent window is limited to 32 frames, the event bank has a capacity of $B = 16$, and each event retains at most two representative frames. For verification-guided event memory, we set $\delta_g = \delta_c = 0.1$ and $N_{\min} = 2$. The promotion score uses $(\alpha_s, \alpha_n) = (0.8, 0.2)$ with a promotion threshold of $\tau_p = 0.75$, while the retention score uses $(\beta_s, \beta_a, \beta_n, \beta_l) = (0.5, 0.2, 0.2, 0.1)$. We use greedy decoding for all experiments. All experiments are conducted on a single NVIDIA A800 GPU.

D.2 Results on LongVideoBench

LongVideoBench evaluates long-context video understanding by requiring models to retrieve and reason about events distributed throughout extended videos. As shown in Table 4, StreamBank improves the MMDuet2 base model on all 17 task categories and raises the average score from 50.6 to 56.5, a gain of 5.9 points. The improvements are especially substantial on T3E ($+11.0$), SOS ($+9.9$), S2E ($+9.7$), and T2E ($+9.2$), indicating stronger reasoning across diverse temporal and event relations. These consistent gains show that Speculation-Guided Event Memory does not merely reduce the historical context: by prioritizing evidence that contributes to target-model predictions, it preserves informative events and limits interference from redundant history, thereby improving long-range video understanding under a fixed memory budget.

D.3 Results on MLVU

We further evaluate StreamBank on the MLVU long-video understanding benchmark. As shown in Table 5, StreamBank achieves an M-Avg of 64.10, improving over the MMDuet2 base model by 4.31 points and obtaining the best performance on Topic Reasoning, Needle Question Answering, Plot Question Answering, and Action Order. The gains are particularly pronounced on tasks that require retrieving localized information or integrating evidence across distant moments. These results demonstrate that Speculation-Guided Event Memory effectively preserves historical information relevant to subsequent predictions under a bounded context budget, thereby improving long-video understanding.

D.4 Results on LVBench

As shown in Table 6, StreamBank improves the overall score of MMDuet2 from 40.6 to 44.0, yielding a substantial gain of 3.4 points. The pronounced gains in ER and EU demonstrate that StreamBank more effectively preserves entity- and event-related evidence from long video histories, while the improvements in TG and Rea indicate stronger cross-temporal information association and reasoning. Although its KIR score decreases slightly by 0.6 points, StreamBank outperforms MMDuet2 on most capability dimensions and achieves a clear overall improvement. These results demonstrate that Speculation-Guided Event Memory reduces interference from redundant history while retaining video evidence that is more valuable for subsequent predictions under a fixed memory capacity.

Table 5: Experimental results on MLVU.

Method	Holistic		Single Detail		Multi Detail			M-Avg
	TR	AR	NQA	ER	PQA	AO	AC	
MovieChat	29.5	25.0	24.2	24.7	25.8	28.6	22.8	25.8
Time-Chat	23.1	27.0	24.5	28.4	25.8	24.7	32.0	30.9
MA-LMM-7B	51.9	35.5	43.1	38.9	35.8	25.1	24.3	36.4
Video-ChatGPT-7B	26.9	24.0	40.3	42.0	29.9	25.1	31.1	31.3
LLaMA-VID	50.8	34.5	30.1	32.7	32.5	23.9	27.8	33.2
LLaVA-Mini	76.0	50.0	44.5	37.5	49.0	24.3	18.4	42.8
Video-LLaVA-7B	71.6	57.0	53.2	45.2	48.4	20.1	**35.9**	47.3
LongVA-7B	82.2	58.5	70.4	48.6	68.1	41.7	25.2	56.4
VideoScan	81.8	64.0	67.9	**62.5**	64.9	44.0	31.6	61.3
MMDuet2†	79.85	**74.50**	62.25	59.09	63.82	55.21	23.79	59.79
StreamBank	**84.03**	67.50	**75.21**	57.10	**73.47**	**59.85**	31.55	**64.10**

Table 6: Results on LVBench.

Method	ER	EU	KIR	TG	Rea	Sum	Overall
TimeChat	21.9	21.7	25.9	22.7	25.0	24.1	22.3
MovieChat	21.3	23.1	25.9	22.3	24.0	17.2	22.5
Gemini 1.5 Pro	32.1	30.9	39.3	31.8	27.0	32.8	33.1
GLM-4V-Plus	39.9	35.8	34.8	37.7	40.0	32.8	38.3
InternVL2-40B	37.4	39.7	**43.4**	31.4	42.5	41.4	39.6
Qwen2-VL-72B	38.0	41.1	38.3	41.4	**46.5**	46.6	41.3
TimeMarker	42.8	39.1	34.9	38.7	38.2	**48.8**	41.3
MMDuet2	40.2	39.0	38.1	40.5	44.3	32.8	40.6
StreamBank	**45.1**	**42.0**	37.5	**41.4**	46.3	32.8	**44.0**

E VISUALIZATION OF STREAMBANK

To provide a more intuitive analysis of StreamBank, we visualize the historical information retained by different memory policies in three episodic-memory examples from OVO-Bench. As shown in Figures 4, 5, and 6, **R** denotes the recent-frame window, **F** denotes the FIFO policy, **V** denotes the Visual Similarity policy, and **O** denotes our StreamBank. Under a fixed memory budget, StreamBank preserves historical evidence that remains useful for subsequent questions and consequently produces correct predictions. These qualitative examples further show that *Speculation-Guided Event Memory* can retain useful long-range evidence within a bounded memory capacity, enabling long-range video understanding without preserving the complete visual history.

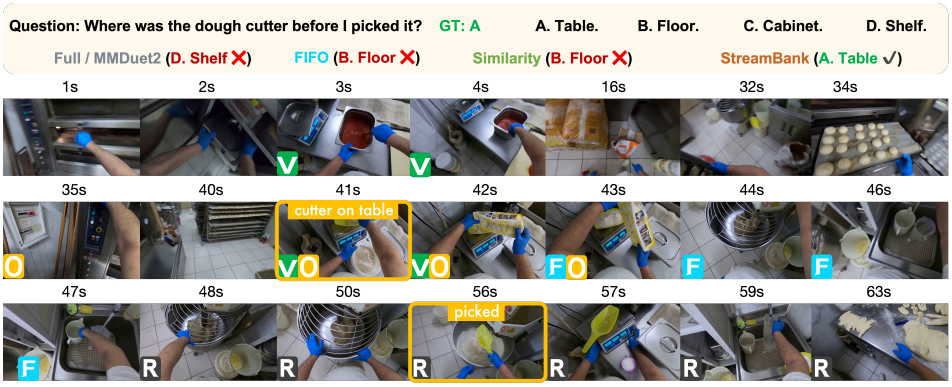

Figure 4: Qualitative comparison on OVO-Bench EPM example #88.

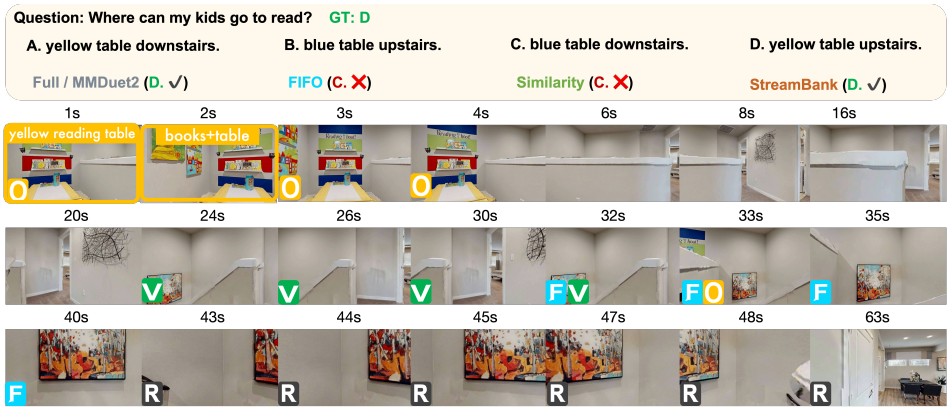

Figure 5: Qualitative comparison on OVO-Bench EPM example #209.

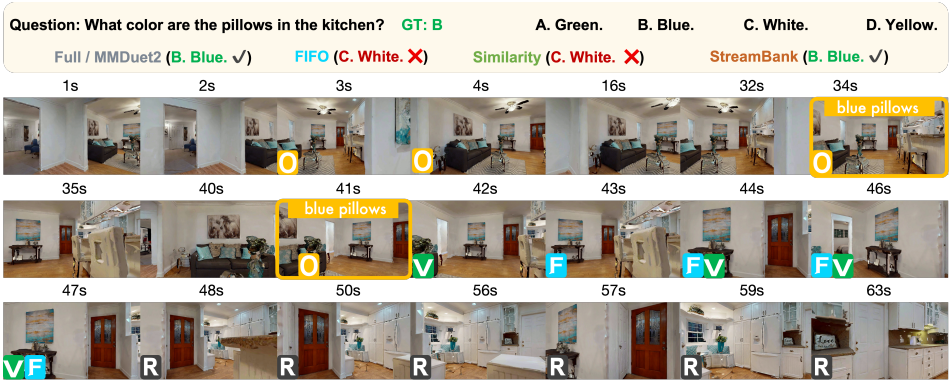

Figure 6: Qualitative comparison on OVO-Bench EPM example #201.