StreamRAG: Enhancing Real-Time Video Understanding with Retrieval Augmentation

Supplementary Material

In these supplementary materials, we provide the following:

- Detailed experimental supplements about our StreamRAG (Section 1);
- Prompt examples demonstration (see Section 2);

1. EXPERIMENT

1.1. Supplementary RAG Methods Comparison.

VideoRAG [5] The current VideoRAG method, based on the VideoMME dataset, achieves notable performance in conventional video scenarios by incorporating expensive knowledge as auxiliary information. However, our experiments on streaming video datasets reveal its suboptimal effectiveness. Further analysis identifies two critical factors contributing to this performance gap: First, the contribution of speech information varies significantly across scenarios. While VideoRAG's strong performance on VideoMME heavily relies on speech-based knowledge, its utility diminishes in streaming video contexts, where audio plays a less critical role in task completion. Second, the CLIP-based frame selection mechanism struggles in streaming settings, particularly for first-person videos containing excessive irrelevant environmental objects. This leads to unreliable relevance filtering under threshold-based approaches. Even when replacing thresholding with a fixed top-8 frame selection strategy, performance gains remain marginal, indicating fundamental limitations in adapting conventional VideoRAG to streaming video scenarios.

StreamChat-RAG [11] While the authors [11] do not explicitly use the term "RAG" and instead emphasize a "dynamic memory system", we argue that the work essentially constitutes an innovative extension of the RAG framework tailored for streaming video scenarios. Therefore, we have redefined and named it StreamChat-RAG. To our knowledge, StreamRAG represents the first explicitly proposed RAG concept for streaming video. Unlike conventional RAG approaches, StreamChat-RAG constructs a tree-structured knowledge index that is progressively updated. It employs query-caption similarity matching to select relevant video segments and integrates both short-term and long-term memory into the response generation process. However, the current implementation exhibits **several limitations** in streaming video contexts: the fixed frame-rate update mechanism fails to adequately capture the temporal dynamics inherent in streaming data, while the computationally intensive caption extraction process creates a processing bottleneck. Moreover, the reliance solely on query similarity for memory retrieval overlooks the critical time-sensitive nature of streaming video content, where temporal coherence often carries essential contextual information. These constraints collectively hinder the framework's ability to build a high-performance RAG library for streaming video—limitations that our framework systematically addresses.

Table 1. **Quantitative results in StreamBench.**

Method	FPS	Fr.	Sco.	Acc.
Human performance	-	-	4.03	79.4
GPT-4o [4]	-	50	3.70	71.0
GPT-4o [4]	-	35	3.64	69.8
GPT-4o-mini [4]	-	35	3.17	59.1
Instruct-tuning				
Video-LLava [8]	-	8	2.81	48.9
LLama-VID [7]	-	180	2.94	51.2
LLava-NExT [6]	-	8	2.65	46.2
LLava-Hound [14]	-	8	3.12	54.7
LongVA [13]	-	8	3.05	52.4
MiniCPM-v2.6 [3]	-	8	2.97	56.6
Training-free				
MovieChat [9]	-	32	2.07	35.3
FreeVA [10]	-	4	3.10	56.3
Streaming				
Video-online [2]	5	-	3.11	56.4
Flash-VStream [12]	1	-	2.89	52.10
Qwen2vl [1]	1	-	2.95	55.05
Qwen2vl+ours [1]	1	-	**3.07**	**57.85**

1.2. Supplementary Evaluation on Open Datasets.

To further validate the effectiveness of our approach beyond the benchmark evaluations presented in the paper, we conducted additional experiments on StreamBench [11], an open-ended dataset for streaming video question answering. Following the same evaluation protocol as described in the original work, we employed the LLaMA-3 model to assess response quality through both scoring and accuracy (Acc) calculations. Specifically, we measure semantic similarity in individual conversations by having LLaMA-3 assign a semantic correctness score (Sco.) on a scale of [0, 5], where higher scores indicate responses that more precisely align with the expected answers. As shown in Table 1,

Table 2. **Performance on OVOBench about Advanced Models.**

Model	Setting	Real-Time Visual Perception						Avg.	Backward Tracing			Avg.
		OCR	ACR	ATR	STU	FPD	OJR		EPM	ASI	HLD	
Qwen3-VL	1fps	77.85	62.39	72.41	56.18	65.35	61.41	65.93	54.88	68.92	12.37	45.39
Qwen3-VL+ours	1fps	81.88	60.55	71.55	55.62	70.30	63.04	67.16	54.21	66.89	20.43	47.18
InternVL3.5	64	73.83	62.39	70.69	57.30	72.28	63.59	66.68	54.88	62.16	9.14	42.06
InternVL3.5+ours	64	82.55	66.06	72.41	56.18	72.28	66.85	69.39	52.86	62.84	13.98	43.23

our method achieves a consistent improvement of 2 percentage points compared to baseline approaches. This performance gain demonstrates that our approach not only excels in structured benchmark evaluations but also exhibits strong generalization capabilities in open-ended scenarios, where the diversity of possible correct answers presents additional challenges for video understanding and response generation. The improved performance on StreamBench particularly highlights our method's enhanced ability to comprehend dynamic visual content and generate semantically appropriate responses in more realistic, unconstrained settings.

Query Instant Judgment Prompt

Role
You are a **real-time video analyzer** scoring how strictly a query requires ONLY current video/audio (1-3 seconds of content).
Output Format
JSON with keys: { "real_time_score": 0.0-1.0, # 0=non-realtime 1=purely realtime "reason": "internal_analysis" # for debugging only }
Scoring Rules:
Scores range from 0.0 (non real-time) for queries with time references ("before"/"next"), comparisons ("is it faster than earlier?"), or cross-frame reasoning ("where is he walking?"), to 1.0 (purely real-time) for single-frame verifiable queries without temporal references. Intermediate values include 0.3 for implicit temporal dependencies (e.g., "is he speaking?") and 0.7 for "now"-based but frame-verifiable queries (e.g., "what number is on screen now?").

1.3. Evaluation on More Advanced Models

This framework encompasses the integration of numerous advanced MLLMs, in Table 2. Their stable performance without any architectural alterations empirically substantiates the plug-and-play nature of our proposed method.

Caption prompt

Role
- You are a detailed video captioning model that generates a comprehensive yet concise description of the visual content based on the provided frames.
Input Instructions
- The input consists of selected frames sampled from a continuous video segment. - Analyze the sequence of frames holistically to infer actions, objects, and context.
Output Format
Produce a single paragraph (3-5 sentences) summarizing: 1. Key Subjects – Identify all visible entities (e.g., people, animals, objects, text/logos) with relevant details (appearance, clothing, tools, etc.). 2. Main Actions – Describe movements, interactions, or changes (e.g., "a person walks toward a table and picks up a cup"). 3. Context & Details – Note the setting (indoor/outdoor), lighting, notable objects, or any recurring elements (e.g., "a cluttered office with monitors").
Focus:
- Be specific: Include colors, spatial relationships ("left/right"), and temporal progression ("then", "while"). - Avoid assumptions: Only describe what is visually evident (no guessing intent or backstory). - Flow naturally: Combine elements into a fluid narrative (e.g., "A woman in a red jacket arranges papers on a desk, then turns to speak to a man holding a laptop").
Example Output:
"A young man wearing a black t-shirt and jeans stands in a kitchen, slicing vegetables on a cutting board. He pauses to glance at a smartphone placed near a bowl of chopped tomatoes. The room is brightly lit with modern appliances, and a window shows daylight outside. Behind him, a blurred figure walks past carrying a stack of plates."

1.4. Detail of Attributions.

OVOBench. "EPM" represents Episodic Memory, "ASI" represents Action Sequence Identification, "HLD" represents Hallucination Detection, "STU" represents Spatial Understanding, "OJR" represents Object Recognition,

"ATR" represents Attribute Recognition, "ACR" represents Action Recognition, "OCR" represents Optical Character Recognition, "FPD" represents Future Prediction.

StreamingBench. "OP" represents Object Perception. "CR" stands for Causal Reasoning. "CS" denotes Clips Summarization. "EU" signifies Event Understanding. "TR" represents Text-Rich Understanding. "SU" denotes Spatial Understanding. "ACP" refers to Action Perception. "CT" represents Counting. "ACU" refers to Anomaly Context Understanding, "MCU" stands for Misleading Context Understanding.

2. PROMPT

In the supplementary materials, we present the detailed prompts used for generating captions and for evaluating the "transience" score of queries using the large language model (LLM).

References

[1] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023. 1

[2] 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*, pages 18407–18418, 2024. 1

[3] Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, et al. Minicpm: Unveiling the potential of small language models with scalable training strategies. *arXiv preprint arXiv:2404.06395*, 2024. 1

[4] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 1

[5] Soyeong Jeong, Kangsan Kim, Jinheon Baek, and Sung Ju Hwang. Videorag: Retrieval-augmented generation over video corpus. *arXiv preprint arXiv:2501.05874*, 2025. 1

[6] Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv preprint arXiv:2407.07895*, 2024. 1

[7] Yanwei Li, Chengyao Wang, and Jiaya Jia. Llama-vid: An image is worth 2 tokens in large language models. In *European Conference on Computer Vision*, pages 323–340. Springer, 2024. 1

[8] 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. *arXiv preprint arXiv:2311.10122*, 2023. 1

[9] Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, 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*, pages 18221–18232, 2024. 1

[10] Wenhao Wu. Freeva: Offline mllm as training-free video assistant. *arXiv preprint arXiv:2405.07798*, 2024. 1

[11] Haomiao Xiong, Zongxin Yang, Jiazuo Yu, Yunzhi Zhuge, Lu Zhang, Jiawen Zhu, and Huchuan Lu. Streaming video understanding and multi-round interaction with memory-enhanced knowledge. *arXiv preprint arXiv:2501.13468*, 2025. 1

[12] Haoji Zhang, Yiqin Wang, Yansong Tang, Yong Liu, Jiashi Feng, Jifeng Dai, and Xiaojie Jin. Flash-vstream: Memory-based real-time understanding for long video streams. *arXiv preprint arXiv:2406.08085*, 2024. 1

[13] Peiyuan Zhang, Kaichen Zhang, Bo Li, Guangtao Zeng, Jingkang Yang, Yuanhan Zhang, Ziyue Wang, Haoran Tan, Chunyuan Li, and Ziwei Liu. Long context transfer from language to vision. *arXiv preprint arXiv:2406.16852*, 2024. 1

[14] Jiaxing Zhao, Boyuan Sun, Xiang Chen, Xihan Wei, and Qibin Hou. Llava-octopus: Unlocking instruction-driven adaptive projector fusion for video understanding. *arXiv preprint arXiv:2501.05067*, 2025. 1