VideoChat-A1: Thinking with Long Videos by Chain-of-Shot Reasoning

Zikang Wang[1,2,*], **Boyu Chen**[3,4,6,*], **Zhengrong Yue**[1,2,*], **Yi Wang**[2], **Yu Qiao**[2], **Limin Wang**[5,2], **Yali Wang**[3,2,†]

[1]Shanghai Jiao Tong University
[2]Shanghai Artificial Intelligence Laboratory
[3]Shenzhen Key Lab of Computer Vision and Pattern Recognition, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences
[4]School of Artificial Intelligence, University of Chinese Academy of Sciences
[5]Nanjing University
[6]VIVO AI Lab

Abstract

Recent advances in video understanding have been driven by MLLMs. But these MLLMs are good at analyzing short videos, while suffering from difficulties in understanding videos with a longer context. To address this difficulty, several agent paradigms have recently been proposed, using MLLMs as agents for retrieving extra contextual knowledge in a long video. However, most existing agents ignore the key fact that a long video is composed with multiple shots, i.e., to answer the user question from a long video, it is critical to deeply understand its relevant shots like human. Without such insight, these agents often mistakenly find redundant even noisy temporal context, restricting their capacity for long video understanding. To fill this gap, we propose VideoChat-A1, a novel long video agent paradigm. Different from the previous works, our VideoChat-A1 can deeply think with long videos, via a distinct chain-of-shot reasoning paradigm. More specifically, it can progressively select the relevant shots of user question, and look into these shots in a coarse-to-fine partition. By multi-modal reasoning along the shot chain, VideoChat-A1 can effectively mimic step-by-step human thinking process, allowing the interactive discovery of preferable temporal context for thoughtful understanding in long videos. Extensive experiments show that, VideoChat-A1 achieves the state-of-the-art performance on the mainstream long video QA benchmarks, e.g., it achieves 77.0 on VideoMME (w/ subs) and 70.1 on EgoSchema, outperforming its strong baselines (e.g., InternVL2.5-8B and InternVideo2.5-8B), by up to 10.1% and 6.2%. Compared to leading closed-source GPT-4o and Gemini 1.5 Pro, VideoChat-A1 offers competitive accuracy, but only with 7% input frames and 12% inference time on average.

Introduction

Video understanding is an important problem in computer vision (You et al. 2024). With fast development of MLLMs, video understanding has achieved significant progress (OpenAI 2024; Wang et al. 2025). However, most existing MLLMs are good at understanding short videos in seconds, while hardly analyzing longer videos in minutes (or hours) properly. The main challenge lies in feeding a huge amount of frames in long videos to MLLMs. To deal with this problem, several attempts have been proposed by modeling such long multimodal context (OpenAI 2024; Liu et al. 2025a), or equipping MLLMs with token compression (Li et al. 2025; Shu et al. 2024), But their efficiency and effectiveness still need to be further improved in tackling redundant video content, thus blocking their performance on long video understanding benchmarks.

Recent studies have shown that, agent-based approaches are promising for long video understanding (Wang et al. 2024b; Chen et al. 2025; Kugo et al. 2025; Liu et al. 2025b; Zhi et al. 2025; Wang et al. 2024c). Instead of feeding the entire long video with model adaptation, they simplify this challenge by invoking various tools to extract and retrieve relevant information from long videos. However, these approaches lack flexibility to capture complex content changes in a long video, since the retrieval is based on off-the-shelf video knowledge which is fixed after extraction. Alternatively, OpenAI o3 (OpenAI et al. 2025) shows a human-like thinking process on images. Via chain-of-thought reasoning, o3 can deeply understand visual content via progressive interaction with images. However, they ignore that, a long video consists of multiple shots. To answer the user question reliably, it is critical to deeply understand its relevant shots via multi-round reasoning. Without this consideration, these agents often mistakenly find redundant even noisy temporal context, restricting their capacity for long video understanding.

To fill this gap, we propose VideoChat-A1, a novel video agent framework that can progressively think with a long video, via an interactive chain-of-shot reasoning paradigm. Specifically, for each shot reasoning step, we leverage MLLM as core agent of thinking, and invoke tools for selecting relevant shots, dividing these shots into subshots, and reasoning answer with subshots. If the answer is unconfident, it means that the current subshots are insufficient for understanding user question. As a result, VideoChat-A1 will start next-step shot reasoning to further understand user question with finer relevant shots. Through such a distinct chain-of-shot paradigm, our VideoChat-A1 can effectively mimic the thinking process of human, by progressively reasoning on user answer while iteratively looking into relevant video shots, as shown in Fig. 1. Extensive experiments show that, our

[*]Equal contribution.
[†]Corresponding author.

Figure 1: **Motivation.** Direct reasoning models such as GPT-4o (OpenAI 2024) perform global sampling and struggle to focus on key information. Agent-based methods like VideoTree (Wang et al. 2024c) often suffer from incorrect or redundant sampling that leads to noisy captions. In contrast, VideoChat-A1 interactively employs shot perception and reasoning via Chain-of-Shot, which progressively looks into relevant shots through a reflective process for better performance.

VideoChat-A1 achieves the state-of-the-art performance on 4 mainstream benchmarks, e.g., it achieves 77.0 on VideoMME with subset and 70.1 on EgoSchema, outperforming its baselines such as InternVL2.5-8B and InternVideo2.5-8B, by up to 10.1% and 6.2%. Compared to GPT-4o and Gemini 1.5 Pro, VideoChat-A1 offers competitive accuracy, but with 7% input frames and 12% inference time on average.

Related Work

MLLM for Long Video Understanding. MLLMs have demonstrated great potential in the field of video understanding (OpenAI 2024; Li et al. 2023). However, they still face challenges when dealing with long videos that can be minutes or hours in length (Song et al. 2024). Currently, there are two main approaches to addressing this problem. One is to compress visual tokens (Song et al. 2024; Zeng et al. 2024). For example, MA-LLM (He et al. 2024) merges similar tokens, and LLaMA-VID (Li, Wang, and Jia 2025) compresses each frame into context and content tokens to enable longer video input. However, these methods result in a significant loss of visual information. Another approach is to extend the number of processable tokens (Liu et al. 2024; Wang et al. 2024a). Models like LongVILA (Xue et al. 2024; Li et al. 2025) have adopted strategies to increase the token capacity for handling longer videos. However, this method introduces redundant information, consumes more computational resources, and often achieves only moderate effectiveness.

Agent Based Method for Long-context Understanding. Agent mechanisms (Chen et al. 2025; Kugo et al. 2025; Liu et al. 2025b; Ye et al. 2025) have been introduced in video understanding tasks. Currently, Agent methods for long video understanding can be mainly divided into two categories. One is extracting knowledge from long videos by invoking external tools at once and answering questions by retrieving the extracted knowledge. For example, CLIP (Radford et al. 2021) is used for retrieving key frames related to the questions (Zhao et al. 2024; Xie et al. 2023). Memory banks (Fan et al. 2024; He et al. 2024) and search engines (Li et al. 2024b; Chen et al. 2024a) are applied for extracting key information

of the videos. However, this single-pass information extraction often misses critical details or introduces redundant information, which interferes with the model's response. The other approach involves answering questions through a round by round key information search. The multi-round pipeline emphasizes active video exploration and dynamic information acquisition (Wang et al. 2024c,b; Ma et al. 2024), and optimizes the reasoning path through retrieval and reflection mechanism. However, a critical limitation is that most methods either ignore video shots—the fundamental structural units of long-form videos—or segment shots at once without further fine-grained partitioning, which can both miss critical query-relevant details within each shot and introduce redundant or noisy information. For instance, VideoINSTA (Liao et al. 2024) proposes an event-based spatial-temporal reasoning framework, but its spatial and temporal information extraction processes are relatively independent and lack query-driven interactive iterative refinement.

Method

In this section, we introduce our VideoChat-A1. To deeply understand the user question in a long video, we design a distinct **Chain-of-Shot (CoS)** reasoning paradigm, which can progressively think while discovering relevant shots via multi-round shot partition and dialogues. Specifically, it consists of three key steps in each round of shot reasoning, including **Shot Selection**, **Shot Partition**, and **Shot Reflection**. The framework of VideoChat-A1 is shown in Fig. 2.

Video Glance. To start with, it is necessary to determine if the question of the given video needs to dig into the local shots for answering, since some questions may refer to the global content throughout the entire video. In this regard, we introduce a concise video glance step for pre-judging. First, we uniformly sample 4 frames from the entire video to roughly describe the video. Then, we feed these frames along with user question and options into MLLM, justifying whether it is necessary to view the entire video to answer this question. If MLLM considers the question to be global, we uniformly sample 32 frames from the entire video, and feed them into MLLM for answering. If MLLM considers the question to be local, we start chain-of-shot reasoning.

Shot Selection

For conciseness, we describe the i-th shot reasoning step for illustration. Suppose that, we divide the whole video into M shots at the $(i-1)$-th round,

$$\mathcal{S}^{(i-1)} = \{\mathcal{S}_1^{(i-1)}, ..., \mathcal{S}_m^{(i-1)}, ..., \mathcal{S}_M^{(i-1)}\}, \quad (1)$$

As shown in Fig. 2, we first select the candidate shots from the existing ones at the $(i-1)$-th round, in order to further look into these candidates for reasoning at the i-th round.

Key Information Summary. To achieve this goal, we leverage MLLM to summarize key text information $\mathcal{I}^{(i)}$ to describe how to answer the user question, according to the reasoning results in the previous round,

$$\mathcal{I}^{(i)} = MLLM(\mathcal{Q},\ \mathcal{O},\ \mathcal{V}^{(i-1)},\ \mathcal{H}^{(i-1)}), \quad (2)$$

where $\mathcal{Q}$ and $\mathcal{O}$ refers to the user question and answer options. $\mathcal{V}^{(i-1)}$ refers to the video frames sampled from the shots in the historical rounds. Moreover, $\mathcal{H}^{(i-1)}$ refers to historical reasoning information, including key text information $\mathcal{I}^{(i-1)}$, the chosen answer $\mathcal{A}^{(i-1)}$, and the reason why to choose this answer $\mathcal{R}^{(i-1)}$ at round $(i-1)$.

Shot Selection via Retrieval. After obtaining the key text information $\mathcal{I}^{(i)}$, we leverage it as contextual guidance for shot selection. This can be effectively achieved by shot-text retrieval,

$$\mathcal{C}^{(i)} = CLIP(\mathcal{I}^{(i)},\ \mathcal{S}^{(i-1)}), \quad (3)$$

where we leverage our finetuned LongCLIP (Zhang et al. 2024a) as a retriever, and compute the cosine similarities between key information and each shot in $\mathcal{S}^{(i-1)}$. Finally, we select top N shots as candidates for further investigation, $\mathcal{C}^{(i)} = \{\mathcal{C}_1^{(i)}, ...\mathcal{C}_N^{(i)}\}$, where $\mathcal{C}_n^{(i)}$ refers to the n-th selected shot from the shot set $\mathcal{S}^{(i-1)}$. Additionally, since there is one single shot (i.e., the whole video) at the 1st round, we directly use this shot as the candidate at this round.

Shot Partition

After obtaining the candidate shots $\mathcal{C}^{(i)}$, we next further divide each of them into subshots, to look deeper into it for reasoning. In this paper, we introduce a concise shot partition pipeline as shown in Fig. 3. For notation simplicity, we illustrate how to perform it on a candidate shot.

Key Frame Discovery. To discover the subshots in a candidate shot, we start by finding key frames in the subshots. First, we uniformly sample frames from the candidate shot. To reduce temporal redundancy while maintaining computation efficiency, we sample 1 fps in this paper. Second, we extract the features of the sampled frames by CLIP. Third, we perform K-Means (MacQueen 1967) on these features. As a result, each of K clusters roughly reflects a subshot. The choice of K is discussed in detail in the ablation study section. However, frames in each cluster may not be temporally adjacent, since K-Means Clustering does not take temporal order into account. To construct subshots in temporal order, we further find a key frame in each cluster to divide the shot. Specifically, for each cluster, we choose the frame whose feature is closest to the cluster center, as the key frame in this cluster. Finally, we organize K key frames in the temporal order, $\mathcal{K} = \{\mathcal{K}(1), ..., \mathcal{K}(K)\}$, for the candidate shot.

Subshot Boundary Discovery. After finding K key frames, we next identify the boundary frame between two adjacent key frames to construct K subshots. Specifically, we compute $\mathcal{L}_2$ feature distances between each frame $\mathcal{F}$ and its two adjacent key frames. Then we sum over them as a semantic deviation metric for this frame,

$$d = ||\mathcal{F} - \mathcal{K}(k)||_2 + ||\mathcal{F} - \mathcal{K}(k+1)||_2, \quad (4)$$

A higher value of d indicates that this frame lies far from both adjacent key frames, implying a semantic shift between two subshots. Therefore, we select the frame with the maximum d as the boundary frame between two subshots. With such a mechanism, each candidate shot is divided into K subshots, $\mathcal{C}_n^{(i)} = \{\mathcal{S}_n^{(i)}(1), ..., \mathcal{S}_n^{(i)}(K)\}$. Finally, we update the whole shot set by replacing all the candidate shots as the corresponding subshots,

$$\mathcal{S}^{(i)} \leftarrow \left(\mathcal{S}^{(i-1)} \setminus \mathcal{C}^{(i)}\right) \cup \{\mathcal{S}_n^{(i)}(1), ..., \mathcal{S}_n^{(i)}(K)\}_{n=1}^{N}, \quad (5)$$

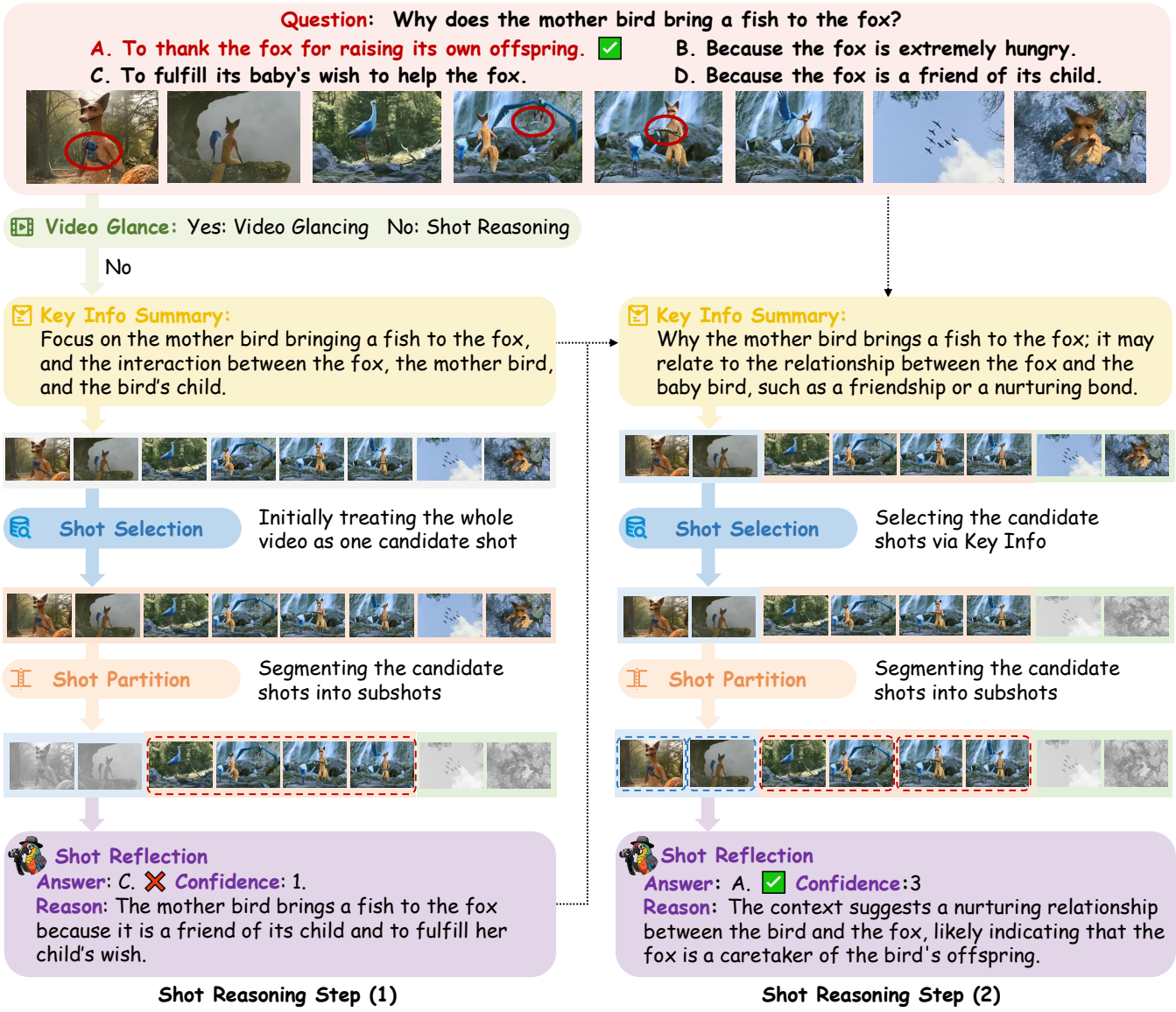

Figure 2: **Framework.** VideoChat-A1 introduces a novel **Chain-of-Shot Reasoning** framework for long video understanding. It progressively refines video analysis through iterative stages of *Shot Selection*, *Shot Partition*, and *Shot Reflection*, leveraging MLLMs to dynamically discover relevant video shots and generate reliable answer.

which is used for shot selection in the next step if necessary.

Shot Reflection

Question Reasoning. After obtaining subshots from the candidate shots, we employ MLLMs to answer the user question based on these finer shot regions. Specifically, as shown in Fig. 4, we leverage MLLM as a dialogue agent to answer the question at round i,

$$\{\mathcal{A}^{(i)}, \mathcal{R}^{(i)}\} = MLLM(\mathcal{Q}, \mathcal{O}, \mathcal{V}^{(i)}), \tag{6}$$

where $\mathcal{Q}$ and $\mathcal{O}$ refer to question and answer options. We sample $\mathcal{T}$ frames from each subshot $\mathcal{S}_n^{(i)}(k)$, where $\mathcal{T}$ is the number of sampled frames, and combine them with $\mathcal{V}^{(i-1)}$ to form $\mathcal{V}^{(i)}$. Finally, given these inputs, MLLM generates answer $\mathcal{A}^{(i)}$, and the reason why to choose this answer $\mathcal{R}^{(i)}$.

Confidence Reflection. Next, MLLM would justify if the answer was reliable to decide if we need to next-round shot reasoning. Hence, we leverage MLLM again to generate the answer confidence, with extra inputs of answer and reason,

$$\mathcal{Z}^{(i)} = MLLM(\mathcal{Q}, \mathcal{O}, \mathcal{V}^{(i)}, \mathcal{A}^{(i)}, \mathcal{R}^{(i)}). \tag{7}$$

We set the confidence level $\mathcal{Z}^{(i)}$ from 0 to 3. If the confidence is higher than 2, we believe that it is a reliable answer as the final output. Otherwise, we believe that it is an unreliable answer and conduct further investigation. Hence, we start shot selection, partition, reflection in the next round. Addi-

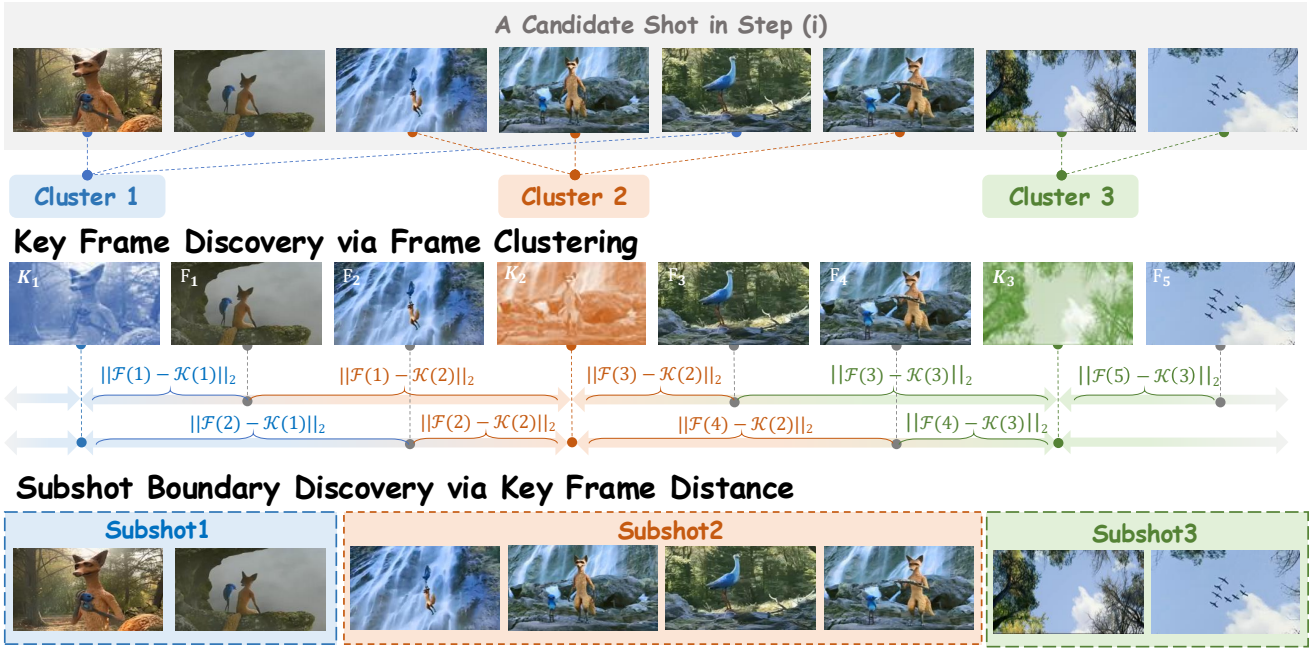

Figure 3: **Shot Partition.** Given a candidate shot at step i, VideoChat-A1 first applies K-Means to obtain K cluster centers for finding key frames. Subsequently, subshots are partitioned based on the feature distance and its adjacent two key frames.

tionally, we set a maximum number of shot reasoning rounds. If VideoChat-A1 still does not get the confident answer at the max step, it will vote the major answer as the final one, according to the answers of all the rounds. Via exploit relevant shots in a coarse-to-fine manner, our VideoChat-A1 allows to leverage a chain of shots for deep thinking on long videos, and thus progressively and interactively discover the reliable shot context to answer user question.

Experiment

Experimental Setup

We evaluate VideoChat-A1 on four long-video question-answering benchmarks: EgoSchema (Mangalam, Akshulakov, and Malik 2023), LongVideoBench (Wu et al. 2024), MLVU (Zhou et al. 2024), and Video-MME (Fu et al. 2024). As baselines, we use Qwen2.5-VL-7B, InternVL2.5-8B, and InternVideo2.5-8B. In the Feature Extraction stage, we sample the video at 1 fps and use pretrained CLIP-ViT-B/32 (Radford et al. 2021) as the feature extractor. In Shot Selection, we sample 16 frames from each shot and use a fine-tuned LongCLIP-B (Zhang et al. 2024a) to calculate shot-text similarity. The details are described in the supplementary documentation. We retrieve shots with similarity scores above the 0.8 threshold. In the first round, we treat the whole video as the candidate shot without selection, and sample $\mathcal{T} = 16$ frames as initial frames. In each subsequent round, we select top 2 shots, divide each shot into 2 subshots, and sample $\mathcal{T} = 8$ frames from each subshot, and add them into frame set. In the Shot Partition stage, the number of clusters K is set to 6 in 1st round and 2 in subsequent rounds. The maximum number of iterations is 3. All experiments are conducted on 2 A800-80GB GPUs.

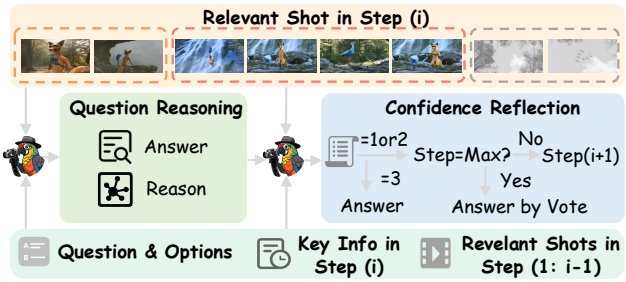

Figure 4: **Shot Reasoning and Shot Reflection.** At each round, VideoChat-A1 performs question-answering reasoning using the relevant shots identified. It then evaluates the confidence level of the response. Then, the system either proceeds to the next step for further refinement or terminates the reasoning process to output the final answer.

Comparison with SOTA

Performance and Efficiency Analysis. As shown in Table 1, we evaluate VideoChat-A1 on four mainstream benchmarks. Specifically, VideoChat-A1 significantly outperforms open-source 7-8B models, comparable to or exceeding that of large open-source or closed-source models. For example, on MLVU, VideoChat-A1 (`InternVideo2.5-8B`) achieves 76.2%, outperforming GPT-4o by 11.6%. Table 2 compares the average frame number and inference time. VideoChat-A1 achieves substantial gains in efficiency. For instance,

Model	EgoSchema	LongVideoBench	MLVU (M-avg)	Video-MME			
				Short	Medium	Long	Average
Closed-Source Model							
GPT-4o(0513) (OpenAI 2024)	72.2	66.7	64.6	80.0 / 82.8	70.3 / 76.6	65.3 / 72.1	71.9 / 77.2
Gemini 1.5 Pro (Reid et al. 2024)	71.1	64.0	-	81.7 / 84.5	74.3 / 81.0	67.4 / 77.4	75.0 / 81.3
Open-Source 72B-78B Model							
LLaVA-OneVision-72B (Li et al. 2024a)	62.0	63.2	68.0	76.7/79.3	62.2/66.9	60.0/62.4	66.3/69.6
VideoLLaMA-2-72B (Cheng et al. 2024)	63.9	-	45.6	69.8/72.0	59.9/63.0	57.6/59.0	62.4/64.7
LLAVA-Video-72B (Zhang et al. 2024b)	65.6	64.9	-	81.4/82.8	68.9/75.6	61.5/72.5	70.6/76.9
Qwen2.5-VL-72B (Bai et al. 2025)	77.9	-	-	80.1/82.2	71.3/76.8	62.2/74.3	71.2/77.8
InternVL-2.5-78B (Chen et al. 2024b)	-	63.6	75.7	82.8/83.2	70.9/74.1	62.6/64.8	72.1/74.0
InternVL-3-78B (Zhu et al. 2025)	-	65.7	79.5	-	-	-	72.7 /75.7
Open-Source 7B-8B Model							
ShareGPT4Video-8B (Chen et al. 2024a)	-	39.7	46.4	48.3/53.6	36.3/39.3	35.0/37.9	39.9/43.6
VideoChat2-7B (Li et al. 2024c)	56.7	39.3	47.9	48.3/52.8	37.0/39.4	33.2/39.2	39.5/43.8
LongVU-7B (Shen et al. 2024)	67.6	-	65.4	-	-	-/59.5	-/60.6
LLaVA-Video-7B (Zhang et al. 2024b)	57.3	58.2	70.8	-	-	-	63.3/69.7
Qwen2.5-VL-7B (Bai et al. 2025)	66.7	55.6	-	-	-	-	63.3/69.0
InternVideo-2.5-8B (Wang et al. 2025)	63.9	60.6	72.8	-	-	-	65.1/-
InternVL-2.5-8B (Chen et al. 2024b)	-	60.0	68.9	-	-	-	64.2/66.9
InternVL-3-8B (Zhu et al. 2025)	-	58.8	71.4	-	-	-	66.3/68.9
Agent Based Model							
VideoAgent (Wang et al. 2024b)	54.1	-	-	-	-	-	-
VideoTree (**GPT-4**) (Wang et al. 2024c)	61.1	-	-	-	-	-	-
DrVideo (Ma et al. 2024)	61.0	-	-	-	-	-	51.7/71.7
T* (Ye et al. 2025)	-	-	-	61.0/-	66.6/-	77.5/-	68.3/-
VideoMind (Liu et al. 2025b)	-	-	64.4	-	-	49.2/-	58.2/-
VideoMultiAgents (Kugo et al. 2025)	68.0	-	-	-	-	-	-
VideoRAG-72B (Luo et al. 2024)	-	65.4	73.8	81.1/-	72.9/-	73.1/-	75.7/-
VideoChat-A1 (**Qwen2.5-VL-7B**)	70.7	64.2	71.9	78.0/81.2	73.1/77.4	64.3/70.9	71.8/76.5
VideoChat-A1 (**InternVL2.5-8B**)	72.1	65.2	75.1	80.8/81.4	72.3/76.8	63.7/72.6	72.3/77.0
VideoChat-A1 (**InternVideo2.5-8B**)	70.1	65.4	76.2	81.4/82.4	72.8/76.7	65.0/71.2	72.9/76.8

Table 1: Comparison with closed-source, open-source, and agent based model on four Long Video Understanding Tasks. The Video-MME results are presented in the format "w/o subs / w/ subs".

VideoChat-A1 (InternVL2.5-8B) reduces frame usage by over 90% and inference time by more than 85% compared to GPT-4o, Gemini 1.5 Pro, and Qwen2.5-VL-72B. Despite these significant savings, our method maintains competitive performance across benchmarks. In contrast to small open-source models, VideoChat-A1 not only delivers significantly better performance but also requires fewer video frames and achieves faster or comparable inference.

Ablation Study

Ablation study of Video Glance. In this study, we conducted an ablation study to verify the impact of the video glance. Table 3 shows that the model with the video glance achieves higher scores on all the tested tasks. The improvements proving the effectiveness of video glance.

Ablation on Shot Partition and Chain-of-Shot. In Table 4, we evaluate the impact of Shot Partition (SP) and Chain-of-Shot (CoS) on VideoChat-A1 (Qwen2.5-VL-7B). The EgoSchema and LongVideoBench are abbreviated as Ego and LVbench in the table because of the limited space. In the setup without SP, we evenly divided the entire video into six shots on average in the 1st round. In subsequent rounds, for the selected shots that need to be expanded, they are directly partitioned into two shots on average. Regarding the setup without CoS, we removed the CoS mechanism. Instead, the model samples additional frames from the video and incorporates them into the existing frame set from the previous round. Then we conducted a vote to get the final response. Individually, enabling SP or CoS boosts performance. Their combined use yields the best results. For example, on MLVU, the score rises from 66.2 (with neither) to 71.9 (both enabled).

Ablation on Shot Partition Methods. As shown in Table 5, we evaluate the impact of three methods of shot partition across multiple benchmarks. Cluster Partition refers to the clustering approach used in VideoTree (Wang et al. 2024c), where all frames within each cluster are treated as a single shot, even though they may not form a continuous video segment. Average Partition refers to dividing each shot into smaller shots by uniformly splitting it into equal parts. Our proposed Shot Partition method consistently outperforms the other two methods, which demonstrate that partitioning videos into temporally coherent shots based on semantic information leads to better understanding of long videos.

Ablation Max Rounds of Reflection. We evaluate the impact of varying the maximum number of reflection rounds. As shown in Table 6, increasing the number of rounds con-

Model	Frames	Inference Time	EgoSchema	LongVideoBench	MLVU	VideoMME
GPT-4o(OpenAI 2024)	384	134.4s	72.2	66.7	64.6	71.9/77.2
Gemini-1.5-Pro(Reid et al. 2024)	568	198.8s	71.1	64.0	-	75.0/81.3
Qwen2.5-VL-72B(Bai et al. 2025)	768	122.4s	77.9	-	-	71.2/77.8
Qwen2.5-VL-7B(Bai et al. 2025)	512	24.3s	66.7	55.6	-	63.3/69.7
InternVL-2.5-8B(Chen et al. 2024b)	64	12.4s	-	60.0	68.9	64.2/66.9
InternVideo-2.5-8B(Wang et al. 2025)	512	33.8s	63.9	60.6	72.8	65.1/-
VideoChat-A1 (`Qwen2.5-VL-7B`)	42.0	18.6s	70.7	64.2	71.9	71.8/76.5
VideoChat-A1 (`InternVL2.5-8B`)	35.9	14.7s	72.1	65.2	75.1	72.3/77.0
VideoChat-A1 (`InternVideo2.5-8B`)	41.3	28.4s	70.1	65.4	76.2	72.9/76.8

Table 2: Comparison on Average Frame Number and Inference Time on Datasets. Inference Time includes the average time for video frame extraction, clustering, and the time required for the MLLM to answer all queries.

7/8B Agent	Video Glance	Ego	LVBench	MLVU	VideoMME
VideoChat-A1 (Qwen2.5-VL-7B)	✗	69.4	63.4	70.0	70.4/74.9
	✓	70.7	64.2	71.9	71.8/76.5
VideoChat-A1 (InternVL2.5-8B)	✗	70.8	64.0	73.9	70.6/75.5
	✓	72.1	65.2	75.1	72.3/77.0
VideoChat-A1 (InternVideo2.5-8B)	✗	69.3	64.3	74.5	71.8/74.9
	✓	70.1	65.4	76.2	72.9/76.8

Table 3: Ablation on the Video Glance in **VideoChat-A1**.

SP	CoS	Ego	LVBench	MLVU	VideoMME
✗	✗	67.1	56.4	66.2	63.4/69.8
✓	✗	67.5	61.1	68.5	67.8/73.9
✗	✓	69.1	57.7	67.0	65.8/70.8
✓	✓	**70.7**	**64.2**	**71.9**	**71.8/76.5**

Table 4: Ablation of Shot Partition (SP) and Chain-of-Shot (CoS) in **VideoChat-A1** (`Qwen2.5-VL-7B`).

Method of Shot Partition	Ego	LVBench	MLVU	VideoMME
Cluster Partition	68.3	58.4	67.7	67.8/72.8
Average Partition	69.1	57.7	67.0	65.8/70.8
Shot Partition	**70.7**	**64.2**	**71.9**	**71.8/76.5**

Table 5: Ablation of different methods of shot partition in **VideoChat-A1** (`Qwen2.5-VL-7B`).

Max Rounds	Ego	LVBench	MLVU	VideoMME
1	67.3	60.1	66.2	66.8/73.4
2	70.0	62.5	69.7	69.6/76.0
3	70.7	64.2	71.9	71.8/76.5
4	70.9	64.5	72.1	72.0/76.9

Table 6: Ablation over maximum iteration rounds in **VideoChat-A1** (`Qwen2.5-VL-7B`).

Number of Init Shots	Ego	LVBench	MLVU	VideoMME
3	70.1	63.9	70.8	71.2/75.7
6	**70.7**	**64.2**	**71.9**	**71.8/76.5**
9	69.3	64.0	71.8	71.5/76.2

Table 7: Ablation over number of initialized shots in **VideoChat-A1** (`Qwen2.5-VL-7B`).

sistently leads to performance improvements. In the context of multi-round discussions, increasing the number of discussion rounds tends to enhance the effectiveness. After three rounds of the chain-of-shot process, the results are already quite satisfactory. As the fourth round yields only marginal improvements, we ultimately adopt the three-round setting to balance performance gains and computational cost.

Ablation on Number of Initialized Shots. We evaluate the effect of the number of initial shots, which is set to 3, 6, and 9. As shown in Table 7, the 6-shot achieves best accuracies compared with the other two configurations. For the subsequent rounds, considering that iterative process generates many subshots and aiming to balance computational efficiency, we fix clustering number for each subshot at $K = 2$.

Conclusion

In this paper, we introduced **VideoChat-A1**, an agent-based framework for effectively addressing long video understanding tasks. Unlike existing methods that often overlook the intrinsic shot-based structure , our method explicitly employs a **Chain-of-Shot reasoning paradigm**. Specifically, VideoChat-A1 progressively identifies and refines relevant video segments through three key iterative steps: **Shot Selection**, **Shot Partition**, and **Shot Reflection**. This iterative interaction mechanism closely mimics human cognitive processes, allowing the agent to thoughtfully reason and accurately answer the questions. Experimental results demonstrate the effectiveness and efficiency of our method, highlighting the substantial benefits in long video understanding tasks.

Acknowledgements

Supported by Shanghai Artificial Intelligence Laboratory, and the National Key R&D Program of China(NO.2022ZD0160505).

References

Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; Dang, K.; Wang, P.; Wang, S.; Tang, J.; Zhong, H.; Zhu, Y.; Yang, M.; Li, Z.; Wan, J.; Wang, P.; Ding, W.; Fu, Z.; Xu, Y.; Ye, J.; Zhang, X.; Xie, T.; Cheng, Z.; Zhang, H.; Yang, Z.; Xu, H.; and Lin, J. 2025. Qwen2.5-VL Technical Report. arXiv:2502.13923.

Chen, B.; Yue, Z.; Chen, S.; Wang, Z.; Liu, Y.; Li, P.; and Wang, Y. 2025. LVAgent: Long Video Understanding by Multi-Round Dynamical Collaboration of MLLM Agents.

Chen, L.; Wei, X.; Li, J.; Dong, X.; Zhang, P.; Zang, Y.; Chen, Z.; Duan, H.; Lin, B.; Tang, Z.; Yuan, L.; Qiao, Y.; Lin, D.; Zhao, F.; and Wang, J. 2024a. ShareGPT4Video: Improving Video Understanding and Generation with Better Captions.

Chen, Z.; Wang, W.; Cao, Y.; Liu, Y.; Gao, Z.; Cui, E.; Zhu, J.; Ye, S.; Tian, H.; Liu, Z.; et al. 2024b. Expanding Performance Boundaries of Open-Source Multimodal Models with Model, Data, and Test-Time Scaling.

Cheng, Z.; Leng, S.; Zhang, H.; Xin, Y.; Li, X.; Chen, G.; Zhu, Y.; Zhang, W.; Luo, Z.; Zhao, D.; and Bing, L. 2024. VideoLLaMA 2: Advancing Spatial-Temporal Modeling and Audio Understanding in Video-LLMs.

Fan, Y.; Ma, X.; Wu, R.; Du, Y.; Li, J.; Gao, Z.; and Li, Q. 2024. VideoAgent: A Memory-augmented Multimodal Agent for Video Understanding. arXiv:2403.11481.

Fu, C.; Dai, Y.; Luo, Y.; Li, L.; Ren, S.; Zhang, R.; Wang, Z.; Zhou, C.; Shen, Y.; Zhang, M.; Chen, P.; Li, Y.; Lin, S.; Zhao, S.; Li, K.; Xu, T.; Zheng, X.; Chen, E.; Ji, R.; and Sun, X. 2024. Video-MME: The First-Ever Comprehensive Evaluation Benchmark of Multi-modal LLMs in Video Analysis. arXiv:2405.21075.

He, B.; Li, H.; Jang, Y. K.; Jia, M.; Cao, X.; Shah, A.; Shrivastava, A.; and Lim, S.-N. 2024. MA-LMM: Memory-Augmented Large Multimodal Model for Long-Term Video Understanding. arXiv:2404.05726.

Kugo, N.; Li, X.; Li, Z.; Gupta, A.; Khatua, A.; Jain, N.; Patel, C.; Kyuragi, Y.; Ishii, Y.; Tanabiki, M.; Kozuka, K.; and Adeli, E. 2025. VideoMultiAgents: A Multi-Agent Framework for Video Question Answering. arXiv:2504.20091.

Li, B.; Zhang, Y.; Guo, D.; Zhang, R.; Li, F.; Zhang, H.; Zhang, K.; Li, Y.; Liu, Z.; and Li, C. 2024a. LLaVA-OneVision: Easy Visual Task Transfer.

Li, C.; Li, Z.; Jing, C.; Liu, S.; Shao, W.; Wu, Y.; Luo, P.; Qiao, Y.; and Zhang, K. 2024b. SearchLVLMs: A Plug-and-Play Framework for Augmenting Large Vision-Language Models by Searching Up-to-Date Internet Knowledge. arXiv:2405.14554.

Li, K.; He, Y.; Wang, Y.; Li, Y.; Wang, W.; Luo, P.; Wang, Y.; Wang, L.; and Qiao, Y. 2023. VideoChat: Chat-Centric Video Understanding.

Li, K.; Wang, Y.; He, Y.; Li, Y.; Wang, Y.; Liu, Y.; Wang, Z.; Xu, J.; Chen, G.; Luo, P.; et al. 2024c. Mvbench: A comprehensive multi-modal video understanding benchmark.

Li, X.; Wang, Y.; Yu, J.; Zeng, X.; Zhu, Y.; Huang, H.; Gao, J.; Li, K.; He, Y.; Wang, C.; Qiao, Y.; Wang, Y.; and Wang, L. 2025. VideoChat-Flash: Hierarchical Compression for Long-Context Video Modeling. arXiv:2501.00574.

Li, Y.; Wang, C.; and Jia, J. 2025. Llama-vid: An image is worth 2 tokens in large language models.

Liao, R.; Erler, M.; Wang, H.; Zhai, G.; Zhang, G.; Ma, Y.; and Tresp, V. 2024. VideoINSTA: Zero-shot Long Video Understanding via Informative Spatial-Temporal Reasoning with LLMs. *arXiv preprint arXiv:2409.20365*.

Liu, H.; Yan, W.; Zaharia, M.; and Abbeel, P. 2024. World Model on Million-Length Video and Language with RingAttention.

Liu, H.; Yan, W.; Zaharia, M.; and Abbeel, P. 2025a. World Model on Million-Length Video And Language With Blockwise RingAttention. arXiv:2402.08268.

Liu, Y.; Lin, K. Q.; Chen, C. W.; and Shou, M. Z. 2025b. VideoMind: A Chain-of-LoRA Agent for Long Video Reasoning. arXiv:2503.13444.

Luo, Y.; Zheng, X.; Yang, X.; Li, G.; Lin, H.; Huang, J.; Ji, J.; Chao, F.; Luo, J.; and Ji, R. 2024. Video-RAG: Visually-aligned Retrieval-Augmented Long Video Comprehension. arXiv:2411.13093.

Ma, Z.; Gou, C.; Shi, H.; Sun, B.; Li, S.; Rezatofighi, H.; and Cai, J. 2024. DrVideo: Document Retrieval Based Long Video Understanding. arXiv:2406.12846.

MacQueen, J. 1967. Some methods for classification and analysis of multivariate observations.

Mangalam, K.; Akshulakov, R.; and Malik, J. 2023. EgoSchema: A Diagnostic Benchmark for Very Long-form Video Language Understanding. arXiv:2308.09126.

OpenAI; :; El-Kishky, A.; Wei, A.; Saraiva, A.; Minaiev, B.; Selsam, D.; Dohan, D.; Song, F.; Lightman, H.; Clavera, I.; Pachocki, J.; Tworek, J.; Kuhn, L.; Kaiser, L.; Chen, M.; Schwarzer, M.; Rohaninejad, M.; McAleese, N.; o3 contributors; Mürk, O.; Garg, R.; Shu, R.; Sidor, S.; Kosaraju, V.; and Zhou, W. 2025. Competitive Programming with Large Reasoning Models. arXiv:2502.06807.

OpenAI. 2024. GPT-4o System Card. arXiv:2410.21276.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision.

Reid, M.; Savinov, N.; Teplyashin, D.; Lepikhin, D.; Lillicrap, T. P.; and et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Shen, X.; Xiong, Y.; Zhao, C.; Wu, L.; Chen, J.; Zhu, C.; Liu, Z.; Xiao, F.; Varadarajan, B.; Bordes, F.; Liu, Z.; Xu, H.; J. Kim, H.; Soran, B.; Krishnamoorthi, R.; Elhoseiny, M.; and Chandra, V. 2024. LongVU: Spatiotemporal Adaptive Compression for Long Video-Language Understanding.

Shu, Y.; Zhang, P.; Liu, Z.; Qin, M.; Zhou, J.; Huang, T.; and Zhao, B. 2024. Video-XL: Extra-Long Vision Language Model for Hour-Scale Video Understanding.

Song, E.; Chai, W.; Wang, G.; Zhang, Y.; Zhou, H.; Wu, F.; Chi, H.; Guo, X.; Ye, T.; Zhang, Y.; et al. 2024. Moviechat: From dense token to sparse memory for long video understanding.

Wang, X.; Song, D.; Chen, S.; Zhang, C.; and Wang, B. 2024a. LongLLaVA: Scaling Multi-modal LLMs to 1000 Images Efficiently via a Hybrid Architecture.

Wang, X.; Zhang, Y.; Zohar, O.; and Yeung-Levy, S. 2024b. VideoAgent: Long-form video understanding with large language model as agent.

Wang, Y.; Li, X.; Yan, Z.; He, Y.; Yu, J.; Zeng, X.; Wang, C.; Ma, C.; Huang, H.; Gao, J.; Dou, M.; Chen, K.; Wang, W.; Qiao, Y.; Wang, Y.; and Wang, L. 2025. InternVideo2.5: Empowering Video MLLMs with Long and Rich Context Modeling. arXiv:2501.12386.

Wang, Z.; Yu, S.; Stengel-Eskin, E.; Yoon, J.; Cheng, F.; Bertasius, G.; and Bansal, M. 2024c. VideoTree: Adaptive Tree-based Video Representation for LLM Reasoning on Long Videos.

Wu, H.; Li, D.; Chen, B.; and Li, J. 2024. LongVideoBench: A Benchmark for Long-context Interleaved Video-Language Understanding. arXiv:2407.15754.

Xie, T.; Zhou, F.; Cheng, Z.; Shi, P.; Weng, L.; Liu, Y.; Hua, T. J.; Zhao, J.; Liu, Q.; Liu, C.; Liu, L. Z.; Xu, Y.; Su, H.; Shin, D.; Xiong, C.; and Yu, T. 2023. OpenAgents: An Open Platform for Language Agents in the Wild. arXiv:2310.10634.

Xue, F.; Chen, Y.; Li, D.; Hu, Q.; Zhu, L.; Li, X.; Fang, Y.; Tang, H.; Yang, S.; Liu, Z.; et al. 2024. Longvila: Scaling long-context visual language models for long videos.

Ye, J.; Wang, Z.; Sun, H.; Chandrasegaran, K.; Durante, Z.; Eyzaguirre, C.; Bisk, Y.; Niebles, J. C.; Adeli, E.; Fei-Fei, L.; Wu, J.; and Li, M. 2025. Re-thinking Temporal Search for Long-Form Video Understanding. arXiv:2504.02259.

You, Z.; Wen, Z.; Chen, Y.; Li, X.; Zeng, R.; Wang, Y.; and Tan, M. 2024. Towards Long Video Understanding via Fine-detailed Video Story Generation. arXiv:2412.06182.

Zeng, X.; Li, K.; Wang, C.; Li, X.; Jiang, T.; Yan, Z.; Li, S.; Shi, Y.; Yue, Z.; Wang, Y.; et al. 2024. Timesuite: Improving mllms for long video understanding via grounded tuning.

Zhang, B.; Zhang, P.; Dong, X.; Zang, Y.; and Wang, J. 2024a. Long-CLIP: Unlocking the Long-Text Capability of CLIP.

Zhang, Y.; Wu, J.; Li, W.; Li, B.; Ma, Z.; Liu, Z.; and Li, C. 2024b. Video Instruction Tuning With Synthetic Data. arXiv:2410.02713.

Zhao, J.; Zu, C.; Xu, H.; Lu, Y.; He, W.; Ding, Y.; Gui, T.; Zhang, Q.; and Huang, X. 2024. LongAgent: Scaling Language Models to 128k Context through Multi-Agent Collaboration.

Zhi, Z.; Wu, Q.; shen, M.; Li, W.; Li, Y.; Shao, K.; and Zhou, K. 2025. VideoAgent2: Enhancing the LLM-Based Agent System for Long-Form Video Understanding by Uncertainty-Aware CoT. arXiv:2504.04471.

Zhou, J.; Shu, Y.; Zhao, B.; Wu, B.; Xiao, S.; Yang, X.; Xiong, Y.; Zhang, B.; Huang, T.; and Liu, Z. 2024. MLVU: A Comprehensive Benchmark for Multi-Task Long Video Understanding.

Zhu, J.; Wang, W.; Chen, Z.; Liu, Z.; Ye, S.; Gu, L.; Duan, Y.; Tian, H.; Su, W.; Shao, J.; Gao, Z.; Cui, E.; Cao, Y.; Liu, Y.; Wei, X.; Zhang, H.; Wang, H.; Xu, W.; Li, H.; Wang, J.; Chen, D.; Li, S.; He, Y.; Jiang, T.; Luo, J.; Wang, Y.; He, C.; Shi, B.; Zhang, X.; Shao, W.; He, J.; Xiong, Y.; Qu, W.; Sun, P.; Jiao, P.; Lv, H.; Wu, L.; Zhang, K.; Deng, H.; Ge, J.; Chen, K.; Wang, L.; Dou, M.; Lu, L.; Zhu, X.; Lu, T.; Lin, D.; Qiao, Y.; Dai, J.; and Wang, W. 2025. InternVL3: Exploring Advanced Training and Test-Time Recipes for Open-Source Multimodal Models. arXiv:2504.10479.