VCA: Video Curious Agent for Long Video Understanding

Zeyuan Yang†
University of Massachusetts, Amherst
yangzeyuan@umass.edu

Delin Chen†
University of Massachusetts, Amherst
delinchen@umass.edu

Xueyang Yu
University of Massachusetts, Amherst
xueyangyu@umass.edu

Maohao Shen
Massachusetts Institute of Technology
maohao@mit.edu

Chuang Gan
University of Massachusetts, Amherst
ganchuang1990@gmail.com

Abstract

Long video understanding poses unique challenges due to their temporal complexity and low information density. Recent works address this task by sampling numerous frames or incorporating auxiliary tools using LLMs, both of which result in high computational costs. In this work, we introduce a curiosity-driven video agent with self-exploration capability, dubbed as "VCA". Built upon VLMs, VCA autonomously navigates video segments and efficiently builds a comprehensive understanding of complex video sequences. Instead of directly sampling frames, VCA employs a tree-search structure to explore video segments and collect frames. Rather than relying on external feedback or reward, VCA leverages VLM's self-generated intrinsic reward to guide its exploration, enabling it to capture the most crucial information for reasoning. Experimental results on multiple long video benchmarks demonstrate our approach's superior effectiveness and efficiency.

1. Introduction

There is a growing demand for developing long-form video understanding techniques [48, 69] capable of extracting valuable information from extended visual narratives, such as surveillance footage, documentaries, and instructional videos. Analyzing these videos [53, 71], which can range from minutes to hours, requires models that can process multimodal data and perform reasoning over extremely long sequences [4, 31, 60].

Previous works commonly utilize two-stream networks to capture spatial and temporal information [18, 19] or introduce 3D operations [8, 57]. Recent approach [32, 75] leverages the long-sequence reasoning capabilities of Large Language Models (LLMs) [1, 64] for long-form video question answering, often by converting videos into densely and uniformly sampled frames [33, 38, 55]. However, this approach is usually inefficient due to the inherent imbalance and varying information density across video segments [61, 78]. Some segments are rich in critical details necessary for answering questions, while others contain only trivial information [66]. Uniform sampling [49, 60], which might focus on redundant frames that increase the computational burden and may overlook frames with essential information, often fails to efficiently utilize the video's dynamic structure for question answering [17, 62].

To address this limitation, recent approaches [29, 34, 36, 68] employ complex architectures [63, 85] or specialized preprocessing techniques [44, 65] to improve the efficiency

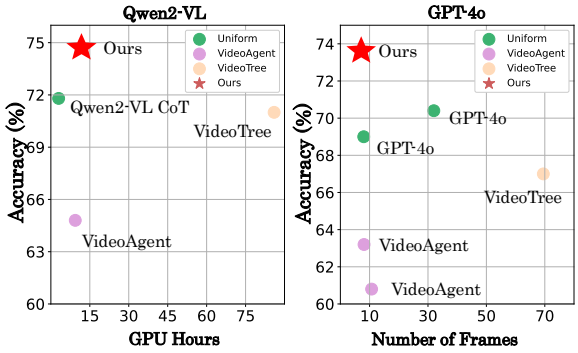

Figure 1. **Accuracy v.s. Computation Cost and Number of Observed Frames.** Compared to baselines, our agent demonstrates superior performance with higher efficiency (observing fewer frames and incurring comparable or lower computational cost) on EgoSchema, using both GPT-4o and Qwen2-VL-72B.

†Equal Contribution

of long video content processing. These methods [62, 66] often convert videos into text by captioning densely sampled frames or incorporating memory modules [5, 17, 39] to store essential information. However, these approaches still incur significant computational costs. For example, captioning an hour-long video with Qwen2-VL-72B at just 1 fps requires over 3 hours on a single H100. Additionally, these methods [12] identify key frames based on image similarity rather than a holistic understanding of the video content, which can result in missing relevant video features. This necessitates the need for an efficient and scalable solution for long-form video understanding.

In this work, we introduce VCA, a curiosity-driven agent inspired by how humans tackle the long video understanding task. Given a question, human is usually intrinsically motivated to identify relevant information in the video without external knowledge. In practise, human tends to build a global understanding of the video, then curiously zoom in to explore specific sections in greater detail. Motivated by this process, we propose a novel video agent framework, dubbed as "VCA", to emulate such behavior. VCA treats the video as an environment to explore, actively navigating its contents and identifying segments that merit deeper analysis. VCA incorporates three essential techniques: (1) Tree-search Exploration: Rather than selecting specific frames [62], VCA uses a tree-search structure to adaptively explore the video segments in a structured way. (2) Reward Model: VCA leverages an intrinsic reward scoring mechanism to guide exploration toward the segments most relevant to the query. (3) Memory Management: VCA employs a fixed memory buffer to store key information and discard irrelevant frames, effectively reducing computational costs and avoiding memory overflow. The key difference between VCA and prior works is outlined in Fig. 2. Extensive experiments and superior performance across multiple benchmarks demonstrate that our proposed VCA framework significantly improves an agent's effectiveness and efficiency in reasoning over extended video content. More concretely, our contributions are threefold,

- VCA is a long video understanding agent that can self-explore the most informative video segments based on its intrinsic reward, without relying on any auxiliary tools.
- VCA is easy to implement, training-free, and is flexible to integrate with any vision-language model.
- As highlighted in Fig. 1, VCA outperforms existing baselines while observing fewer frames and incurring comparable or lower computational cost, demonstrating both effectiveness and efficiency.

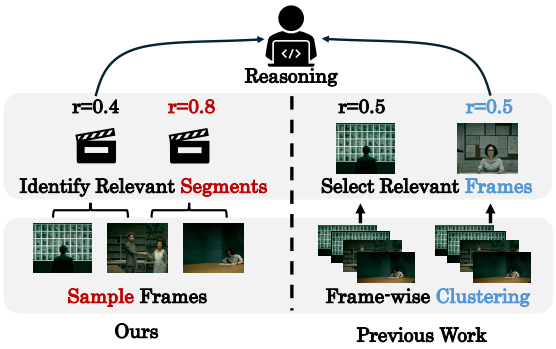

Figure 2. **Comparison of VCA with prior works. Difference-1:** When answering a query, VCA uniformly samples (a few) frames, whereas previous methods (e.g., VideoTree) require captioning each frame for subsequent image retrieval, which can be computationally expensive for long videos. **Difference-2:** previous methods rely on image similarity to locate key frames, potentially losing important video features. In contrast, VCA uses segment-level exploration to identify the most informative segments.

2. Related Work

2.1. Long-context Multi-modal Agents

Large multi-modal models (LMMs) [2, 6, 9, 35, 54] have demonstrated promising performance in addressing complex multimodal tasks [20, 45, 49, 81]. However, these LMMs typically lack self-directed exploration capabilities required for handling more tricky tasks [24, 74]. To bridge this gap, multimodal agents are designed to enhance LMMs with the ability to autonomously make decisions and execute tasks within specific environments [15, 73], such as interpreting visual cues and text prompts to make real-time decisions within websites [28, 76].

A key feature of multimodal agents is their long-context capability [84], i.e., the ability to process and utilize information across extended periods to make informed decisions [26]. An example of this is vision-language model (VLM) agents for video understanding tasks. Despite their advancements, even sophisticated VLM agents struggle to extract crucial information and perform effective reasoning over lengthy inputs [62, 79]. Therefore, efficiently extracting essential information and making informed decisions remains a primary challenge. This work aims to address this challenge by enabling agents to self-explore the environment and self-gather the most relevant information to efficiently solve video-understanding tasks.

2.2. Long Video Understanding

Early deep learning models for video understanding primarily followed two main approaches [52]. One uses two-stream networks [18, 19, 42, 58] to integrate spatial and motion information, while the second focuses on 3D CNNs [8, 57, 72] for spatial-temporal feature extraction. Vision Transformers [14] then introduce self-attention mechanisms

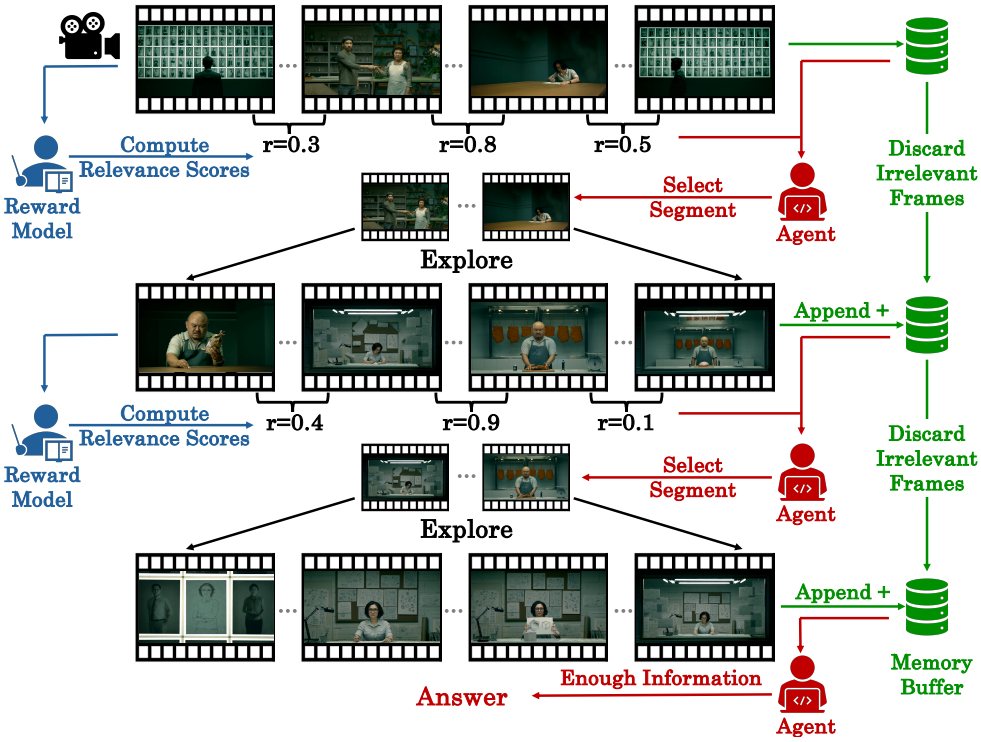

Figure 3. **Pipeline of VCA Framework.** First, the reward model predicts the relevance scores of each segment. Next, the memory buffer is updated by adding new frames and discarding irrelevant ones. Finally, emulating human reasoning, the agent actively selects segments to explore until the correct answer is found.

to handle temporal modeling more effectively [3, 7, 16]. Methods with self-supervised pre-training [50, 56, 86] exhibit transferability across different tasks and achieve more robust video understanding performance.

Multimodal LLM [22, 59, 83] shows strong capabilities in video understanding tasks, but efficiently processing information within a prolonged video environment is the key challenge. Recent approaches often use frame sampling [32, 46, 63, 75, 77] and feature extraction [27, 49, 51, 70] to reduce computational costs. Frame sampling selects a subset of frames, which risks missing essential context in lengthy videos, where transitions and subtle cues may be easily neglected. Additionally, processing entire video sequences as tokens presents significant memory and computational challenges [11, 43, 64, 80].

Recent work has explored auxiliary techniques and modules to improve efficiency, such as pre-extracted content like captions [44, 62], multimodal summaries [40, 65], and combinations of various auxiliary tools [17, 30]. However, these tools introduce additional computational and memory costs. Moreover, some approaches still require traversing the entire video content [17]. Notably, VideoTree [66] requires captioning every frame in the video, making it prohibitively expensive. In contrast, our VCA can solve the task using a single VLM, without relying on any supplementary tools.

3. Method

Inspired by human reasoning, we introduce a self-exploration agent that adaptively navigates video environments. Unlike approaches that depend on dense frame sampling or captioning [62], which require VLMs to process numerous frames, our agent analyzes videos iteratively within a limited context window in a process we call self-exploration. This approach allows essential information to be extracted from long videos in a coarse-to-fine manner.

Specifically, we propose a segment-based tree-search exploration structure. Given a video v and a user query q, the agent π begins by randomly sampling N frames $\{f_v^i\}_{i=1}^N$ to obtain an initial overview of the content in each video segment $v_i \in \{v_i\}_{i=1}^{N+1}$. The agent then decides whether additional exploration is needed to answer the query. If further investigation is necessary, the agent selects a specific segment $s^* \in \{v_i\}_{i=1}^{N+1}$ for deeper exploration until sufficient information is gathered to respond to the query.

Nevertheless, a one-hour movie typically contains approximately 86,400 frames, posing a challenge for current VLM-based agents to efficiently identify the most relevant segment [17, 62, 66]. Moreover, even with a small N, conducting multiple rounds of exploration accumulates a substantial number of frames, leading to high computational costs. To address these issues, we introduce a reward model R to guide the agent's exploration and propose a memory

buffer M to reduce storage and inference costs.

In this section, we first explain how the reward model assists the agent throughout the exploration process (Sec. 3.1). Then we introduce our memory management strategy in Sec. 3.2. Finally, we explain how VCA adaptively explores the video using a segment-based tree structure (Sec. 3.3).

3.1. Reward Model

With the segment-based tree-search exploration, the agent is required to select the most informative video segment for exploration at each step. However, we observe that current VLM-based agents struggle to independently identify optimal exploration paths (see Sec. 4.3). To this end, inspired by the selective attention mechanism in human cognition [13], we incorporate a reward model R to guide the agent's selection process, i.e., the agent will gain a higher reward if it plans to take the action to explore more relevant segment.

Specifically, during the exploration of a current selected segment s^*, the reward model observes a set of frames $\{f^i_{s^*}\}^N_{i=1}$ sampled from s^*. These frames $\{f^i_{s^*}\}^N_{i=1}$ then form finer sub-segments $\{s^*_i = [l^i_{s^*}, l^{i+1}_{s^*}]\}^{N+1}_{i=1}$, where $l^i_{s^*}$ and $l^{i+1}_{s^*}$ represent the indices of the beginning and end frame of the sub-segment, respectively. The reward model aims to generate a reward score for each sub-segment $\{s^*_i\}^{N+1}_{i=1}$ based on its relevance to the user query q. We utilize chain-of-thoughts [67] technique to let the reward model provide a verbal explanation of each segment and then generate a relevant score, i.e., $\{t_{s^*_i}; r_{s^*_i}\}^{N+1}_{i=1}$. Additionally, to maintain consistency across different exploration steps, we incorporate the reward history H_r from previous steps alongside the new segments during evaluation. A simplified prompt template for the reward model is presented here, with detailed prompts in Appendix **??**.

Reward Model Prompt (Simplified)

Given:
H_r, reward score history in previous steps.
$\{f^i_{s^*}\}^N_{i=1}$, newly sampled frames from current segment s^*.
Please first explain each sub-segment $\{s^*_i\}^{N+1}_{i=1}$ and then assign a relevance score of each sub-segment to the user query q.

In practice, rather than training a separate specialized model, we utilize a same VLM-based agent to function as the reward model. While the reward model R and exploration agent π serve distinct purposes, both can be implemented using the same model. In this work, we employ a single VLM-based agent for both roles across different settings, providing a neat and efficient agent framework that is guided by its self-generated intrinsic reward and without relying on auxiliary models or tools.

3.2. Memory Management

As exploration proceeds over multiple rounds, the accumulation of frames can lead to significant computational costs for long videos, even when N is small. However, as illustrated in Fig. 2, they tend to gradually disregard irrelevant frames, especially from earlier steps, retaining only the most crucial information in their working memory for later analysis. Inspired by this, we introduce a fixed-size memory buffer M that stores only the most relevant frames throughout exploration.

Specifically, when the agent π explores a video segment s^*, the associated frames $\{f^i_{s^*}\}^N_{i=1}$ are added to the memory buffer. If the total number of frames exceeds the buffer limit, irrelevant frames are removed based on relevance scores $r_{s^*_i}$ defined in Sec. 3.1. We discard the frames with the lowest scores and only retain the most relevant ones. While these less relevant frames are removed, their visual information is extracted and encoded as text descriptions $\{t_{s^*_i}\}^N_{i=1}$ within the exploration history H_r, allowing the reward model to utilize. This memory management strategy enables the agent to conduct reasoning within a fixed resource limit, regardless of the exploration trajectory's length, thereby preventing excessive memory usage.

3.3. Tree-Search Exploration

Inspired by humans adaptively focusing on different video segments during exploration rather than targeting specific frames directly, our proposed agent VCA treats the video as an unexplored environment, where each segment represents a node on an exploration tree. Given the entire video as the root node, the agent's goal is to efficiently identify an optimal short path leading to a leaf node (segment) containing the most informative content needed to answer the query.

At each step, the agent π receives both textual and visual guidance: a set of candidate segments S, which includes a collection of segments with associated reward scores, and a memory buffer M containing multiple frames. If the agent has gathered sufficient information to answer the user's query q, it directly generates a response a. Otherwise, it continues exploring by selecting a specific segment $s^* \in S$ to expand the tree based on reward scores.

However, due to the limited capacity of current foundational models, these reward scores are not always accurate (see further analysis in Sec. 5.3). Selecting segments based on reward scores in a purely greedy manner can potentially lead to a suboptimal solution. Instead, we use the reward scores as guidance for agent π, allowing it to make the final segment selection independently. In this case, the agent is not forced to select the segment with the highest reward, balancing the exploitation and exploration. More importantly, the candidate set S includes both unexplored

coarse segments from prior steps and fine-grained segments from the current step, enabling the agent to backtrack and select earlier segments if it detects that the current exploration path may be suboptimal. Below are the simplified prompts. The full prompt is provided in Appendix **??**.

Exploration Agent Prompt (Simplified)

Given:
Candidate segments set: S
Frames in Memory buffer: M
Please choose one segment to explore based on the query q. Output answer if you have gathered sufficient information.

The overall framework of our proposed method is summarized in Algo. 1. Leveraging a tree-search exploration technique boosted with a reward model and a memory buffer, our self-exploration agent adaptively and efficiently explores the video like a human. Empirical results in Sec. 4.3 further validate the effectiveness of our tree-search exploration over naive frame selection methods.

4. Experiments

4.1. Experimental Settings

Benchmark. We mainly evaluate VCA on two long-term video question-answering benchmarks. **EgoSchema** [41] is an egocentric video dataset sourced from Ego4D [21], with a duration of 180 seconds. Here we use the validation set of 500 QA pairs. **LVBench** [61] covers about 1,500 samples for evaluation of reasoning abilities from high-quality videos over 30 minutes in length. We also conduct experiments on other relevant benchmark datasets in Appendix **??**.

Baselines. We compare with most relevant baselines to our work, including state-of-the-art open-source Video-LLMs [10, 23, 37, 47, 59, 82], the close-source model GPT-4o [25], and agent-based systems [44, 62, 66]. For video models, we report the results obtained with uniform sampling frames, while for the agent-based systems, we follow their official implementation with GPT-4o [25], consistent with VCA's setup to ensure a fair comparison.

Implementation Details. In this work, we use the same model for the exploration agent and the reward model. Unless otherwise stated, all experiments were performed using the August 2024 version of `gpt-4o` as the base model, leveraging the OpenAI LLM API service[1] with a temperature of 0.5. We set memory buffer sizes set to 8 and 16 frames for EgoSchema and LVBench, respectively. Further implementation details are provided in Appendix **??**.

Algorithm 1 VCA: Video Curious Agent

Require: Video environment v, question q, agent π, reward model R, memory buffer M, candidate segments set S, reward score history H_r, sampling frame number N.
Ensure: Answer a

Initialize selected segment $s^* \leftarrow v$
Initialize candidate segments set $S \leftarrow \emptyset$
Initialize memory buffer $M \leftarrow \emptyset$
Initialize reward score history $H_r \leftarrow \emptyset$
`continue` $\leftarrow$ **true**
while `continue` **do**
  `# Sampling frames & form segments`
  $\{f^i_{s^*}\}_{i=1}^{N}, \{s^*_i\}_{i=1}^{N+1} \leftarrow \text{UniformSample}(s^*)$
  `# Scoring each segment`
  $\{t_{s^*_i}; r_{s^*_i}\}_{i=1}^{N+1} \leftarrow R(\{s^*_i\}_{i=1}^{N+1}; q, \{f^i_{s^*}\}_{i=1}^{N}, H_r)$
  `# Update candidate segment set`
  $S \leftarrow S \cup \{s^*_i; r_{s^*_i}\}_{i=1}^{N+1}$
  `# Update history with thoughts`
  $H_r \leftarrow H_r \cup \{t_{s^*_i}; r_{s^*_i}\}_{i=1}^{N+1}$
  `# Update memory with frames`
  $M \leftarrow \text{UpdateMemory}(M, \{f^i_{s^*}\}_{i=1}^{N}; S)$
  `# Output answer or select segment`
  $a, s \leftarrow \pi(q, S; M)$
  if $a = \emptyset$ **then**
    `# Continue exploration`
    $s^* \leftarrow s$
  else
    `continue` $\leftarrow$ **false**
  end if
end while
return a

4.2. Experimental Results

The experimental results on LVBench and EgoSchema are presented in Tab. 2 and Tab. 1, respectively. By adaptively focusing on critical segments rather than scanning the entire video, our agent not only achieves stronger performance but also incurs less computational cost. With less than 30% of the frames, VCA achieves substantial improvements over directly feeding uniformly sampled frames into GPT-4o. Specifically, we observe performance gains of 3.2% on the EgoSchema subset and 6.6% on LVBench. Significant increases in the *Event Understanding* and *Temporal Grounding* domains further underscore our effectiveness in identifying crucial segments within long videos.

Furthermore, we compare our framework with recently proposed SOTA video agent systems, VideoAgent [62] and VideoTree [66]. For a fair comparison, we re-implement both frameworks using GPT-4o for all reasoning components, consistent with our setup. However, captioning all

[1] https://platform.openai.com/docs/models

Method	Frames	ER	EU	KIR	TG	Rea	Sum	Avg.
TimeChat* [47]	> 96	21.9	21.7	25.9	22.7	25.0	24.1	22.3
LLaVA-NeXT* [83]	32	30.1	31.2	34.1	31.4	35.0	27.6	32.2
InternVL2-40B* [10]	16	37.4	39.7	**43.4**	31.4	42.5	**41.4**	39.8
GLM4V-Plus* [23]	20	39.9	35.8	34.8	37.7	40.0	32.8	38.3
GPT-4o* [25]	64	35.9	30.8	35.5	28.3	33.5	34.5	34.7
VideoAgent† [62]	Avg. 25.5	28.0	30.3	28.0	29.3	28.0	36.4	29.3
VideoTree† [66]	Avg. 103.2	30.3	25.1	26.5	27.7	31.9	25.5	28.8
VCA (ours)	Avg. 20.0	**43.7**	**40.7**	37.8	**38.0**	**46.2**	27.3	**41.3**

Table 1. **Experimental Results on LVBench**. Results of methods marked with * are sourced from LVBench [61]. Methods marked with † indicate implementations where **GPT-4o** serves as the main component. Our agent's memory buffer size is set to **16 frames**. We report the average number of observed frames for agent-based methods, with the best performance highlighted in **bold**.

Method	Frames	Subset
GPT-4o [25]	8	69.0
	32	70.4
VideoAgent† [62]	Avg. 8.1	63.2
	Avg. 10.7	60.8
VideoTree† [66]	Avg. 69.5	67.0
LVNet† [44]	12	68.2
VCA (ours)	Avg. 7.2	**73.6**

Table 2. **Experimental Results on EgoSchema**. Methods marked with † indicate implementations where **GPT-4o** is the main component. The results of LVNet are from [44]. Our agent's memory buffer size is set to **8 frames**. We report the average number of observed frames, with the best performance highlighted in **bold**.

Method	EgoSchema		LVBench	
	Frames	Acc.	Frames	Acc.
VCA (ours)	7.2	**73.6**	20.0	**41.3**
- w/o Reward	7.5	72.6	22.2	39.5
- w/o Tree Search	10.2	69.8	39.9	36.2

Table 3. **Ablation Study on EgoSchema and LVBench**. Both the tree-search exploration and reward model contribute significantly to improvements in performance and efficiency. Detailed ablation study results on LVBench are included in Appendix **??**.

Buffer Size	8	16	32
Acc.	38.9	41.3	**43.2**

Table 4. **Ablation Study of Buffer Size on LVBench.** Increasing the buffer size allows the model to observe more frames during reasoning, leading to improved performance.

frames with GPT-4o is prohibitively expensive. Hence, following the baseline implementations, we use open-source VLMs for captioning. A comparison using the same model for all components is provided in Sec. 5.1. As shown in Tab. 1, VCA outperforms all baselines over 5% on EgoSchema, while requiring processing fewer frames. These results suggest the limitations of dense captioning for video preprocessing tasks, which can incur more computational costs while introducing more redundant information. In contrast, our proposed memory management strategy effectively improves efficiency by only retaining the most crucial information for reasoning. Similar insights can be drawn from results on LVBench in Tab. 2. Additional results are provided in Appendix **??**.

Moreover, we observe that VCA explores approximately three times more frames on LVBench than on EgoSchema. Recall that EgoSchema videos have an average duration of 180 seconds, while videos in LVBench extend to at least 30 minutes. This phenomenon suggests that our proposed tree-search exploration technique actively explores the video to locate the most relevant information and adaptively increases the exploration budget based on task complexity. For example, VCA exhibits both exploitation and exploration: when the agent determines that the current exploration path is suboptimal, it backtracks to previous segments to explore a new path (see Sec. 5.2). These empirical observations further validate the robustness and scalability of our approach, especially in diverse long-video contexts.

4.3. Ablation Study

In this section, we conduct an ablation study to evaluate the components of our agent. Results across both benchmarks are presented in Tab. 3. As illustrated in the table, performance consistently declines when the reward model is removed, highlighting that the agent's inherent reasoning ability alone is insufficient for accurately identifying the most informative segments. This finding emphasizes the effectiveness of incorporating an intrinsic reward to guide the agent's segment selection process. A detailed analysis of our reward model's performance is provided in Sec. 5.3.

Additionally, we observe a significant performance drop

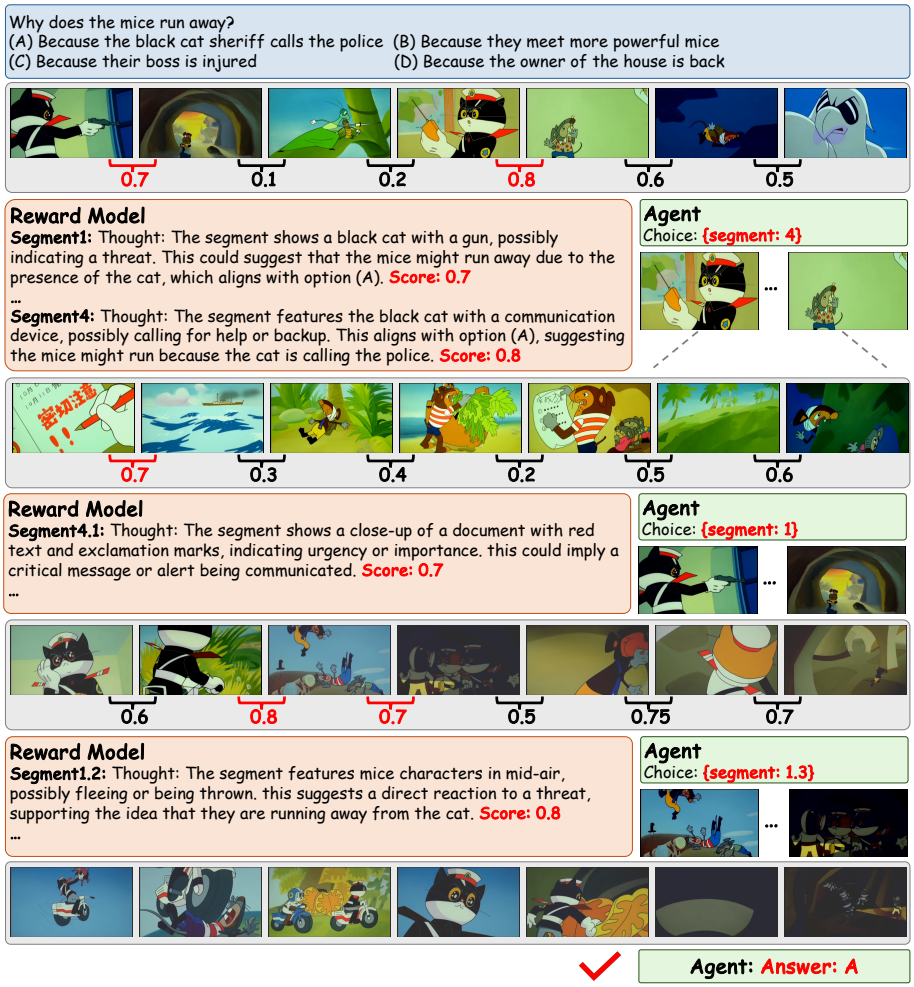

Figure 4. **Exploration Trajectory Example.** The agent generally tends to select video segments with higher reward scores for exploration, yet it can adaptively redirect to earlier segments, effectively balancing exploitation and exploration.

when the tree-search exploration technique is ablated, i.e., the agent directly selects frame indices without the segment-based tree structure. Interestingly, this modification also results in a substantial increase in the average number of frames acquired, especially on LVBench. This indicates that the agent struggles to locate the most crucial information and consequently produces a longer but redundant trajectory. These ablation study results validate the effectiveness of the tree-search exploration framework assisted by the reward model. Additionally, this underscores VCA 's potential to achieve greater efficiency and improved performance with more advanced VLMs and reward models.

Moreover, as shown in Tab. 4, increasing the memory buffer size yields consistently improved performance for VCA, demonstrating that our framework benefits from a broader reasoning window while maintaining effectiveness with fewer frames. Due to limited computation resources, we only tested up to 32 frames; however, the upward trend indicates the potential for gains with additional frames.

5. Analysis

5.1. Can VCA generalize to Open-source Models?

While VCA demonstrates strong performance using GPT-4o as the base model, we also analyze whether these positive results can be extrapolated to open-source models. We implement the VCA framework with Qwen2-VL-72B-Instruct-GPTQ-Int4 [59] as both the exploration agent and reward model, as well as the captioner for baselines. The results in Tab. 5 reveal that, compared to directly feeding uniformly sampled frames into Qwen2-VL, our framework achieves higher accuracy on the EgoSchema benchmark despite using only about 60% of the frames. In contrast, VideoAgent performs significantly worse. Furthermore, our approach outperforms VideoTree with far fewer frames. Notably, VideoTree employs clustering to build a caption similarity-based tree structure for frame selection, requiring caption generation for each frame. This process is computationally expensive, demanding over 60 GPU hours

Method	Frames	Subset
Qwen2-VL	8	70.2
	32	74.2
VideoAgent	Avg. 8.6	65.2
VideoTree	Avg. 85.4	71.0
VCA (ours)	Avg. 16.9	**75.2**
- w/ Image Reward	Avg. 20.9	74.0

Table 5. **Performance of Using Open-source Model on EgoSchema.** Our approach achieves higher accuracy with fewer observed frames, showing consistency improvements over baseline methods. Memory buffer size is set to 16 frames.

on a single H100, as shown in Fig. 1. These results further underscore the robustness of VCA.

5.2. What's the Exploration Behavior of VCA?

Here we present a real exploration trajectory from LVBench to illustrate our agent's decision-making process. As shown in Fig. 4, given a user query, the reward model initially evaluates each segment's relevance based on uniformly sampled frames, and the agent selects the fourth segment, which has the highest relevance score. However, upon further investigation of segment 4, both the reward model and the agent find it relatively irrelevant to the query. Rather than continuing with it, the agent redirects its exploration to segment 1 in the last round. After selecting segment 1, the agent chooses to investigate the third segment, despite the second segment having the highest reward score. This behavior reflects VCA 's ability to adaptively adjust its exploration trajectory based on visual context rather than purely following the reward score, resulting in a more robust system.

We also examine the limitations of VCA by analyzing its failure cases, which fall into the following scenarios: The most common failure occurs when the agent overlooks subtle clues, resulting in an inaccurate response to the user query even after prolonged exploration, particularly for queries requiring specific visual details. Other failure cases include instances where the agent is misled by inaccurate reward scores or fails to provide the correct answer despite observing key frames. Detailed cases and further discussion are provided in the Appendix **??**.

In addition, unlike previous methods, VCA employs a segment-level exploration strategy instead of direct image similarity to select key frames. To further investigate this difference, we evaluate the performance of selecting the segment whose start and end images have the highest visual relevance. As shown in Tab. 5, relying solely on image relevance instead of inferring segment semantics results in a performance drop on EgoSchema, confirming the effectiveness of our segment-level exploration.

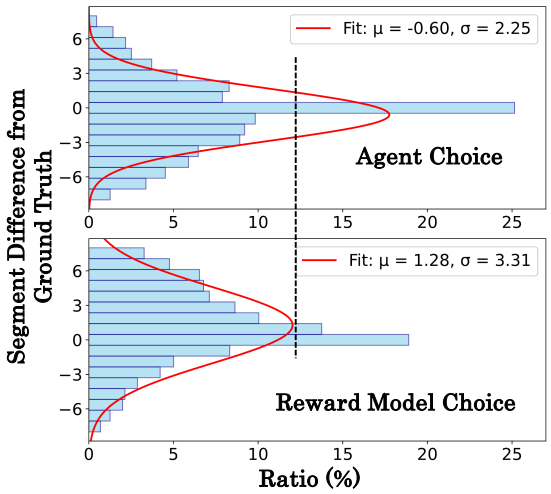

Figure 5. **Segment Selection Accuracy on LVBench.** The y-axis shows segment distance from ground truth; x-axis shows count ratio. Red curves are fitted normal distributions. The reward-guided agent selects segments closer to ground truth than the reward model.

5.3. How Reliable is the Reward Model?

To further assess the capability of our reward model, we visualize the matching accuracy between segments with the highest reward scores and ground truth time references from LVBench. The results are shown at the bottom of Fig. 5. We observe that segment distances follow an approximately normal distribution with a mean of 1.28, indicating that the segment with the highest reward score often closely aligns with the ground truth segment. This validates the reward model's effectiveness in providing meaningful guidance.

Additionally, as discussed in Sec 3.3, we encourage the agent to make independent decisions when selecting segments rather than blindly following reward scores in a purely greedy manner. The benefit of this design choice is demonstrated at the top of Fig. 5, where we present the matching accuracy of the agent's choices. Compared to the greedy approach, the agent's selections result in a sharper distribution of segment distances, with an average distance gap of -0.60, indicating closer alignment with the ground truth. In summary, our reward model assigns reliable relevance scores to segments, while the agent further refines these choices to achieve even more accurate selections.

6. Conclusion

In this work, we propose a curiosity-driven video agent framework for long video understanding tasks with self-exploration capability. VCA achieves state-of-the-art performance and efficiency without relying on auxiliary tools. Appendix **??** shows that improved reward guidance could further boost results, highlighting the potential of synthetic data and specialized reward models for future work.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1

[2] Pravesh Agrawal, Szymon Antoniak, Emma Bou Hanna, Devendra Chaplot, Jessica Chudnovsky, Saurabh Garg, Theophile Gervet, Soham Ghosh, Amélie Héliou, Paul Jacob, et al. Pixtral 12b. *arXiv preprint arXiv:2410.07073*, 2024. 2

[3] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6816–6826, 2021. 3

[4] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023. 1

[5] Ivana Balazevic, Yuge Shi, Pinelopi Papalampidi, Rahma Chaabouni, Skanda Koppula, and Olivier J Hénaff. Memory consolidation enables long-context video understanding. In *Forty-first International Conference on Machine Learning*, 2024. 2

[6] Rohan Bavishi, Erich Elsen, Curtis Hawthorne, Maxwell Nye, Augustus Odena, Arushi Somani, and Sağnak Taşırlar. Fuyu-8b: A multimodal architecture for ai agents, 2023. 2

[7] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding?, 2021. 3

[8] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset, 2018. 1, 2

[9] Xi Chen, Xiao Wang, Lucas Beyer, Alexander Kolesnikov, Jialin Wu, Paul Voigtlaender, Basil Mustafa, Sebastian Goodman, Ibrahim Alabdulmohsin, Piotr Padlewski, et al. Pali-3 vision language models: Smaller, faster, stronger. *arXiv preprint arXiv:2310.09199*, 2023. 2

[10] Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *arXiv preprint arXiv:2404.16821*, 2024. 5, 6

[11] Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, et al. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. *arXiv preprint arXiv:2406.07476*, 2024. 3

[12] Jiwan Chung and Youngjae Yu. Long story short: a summarize-then-search method for long video question answering. *arXiv preprint arXiv:2311.01233*, 2023. 2

[13] Robert Desimone, John Duncan, et al. Neural mechanisms of selective visual attention. *Annual review of neuroscience*, 18(1):193–222, 1995. 4

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021. 2

[15] Zane Durante, Qiuyuan Huang, Naoki Wake, Ran Gong, Jae Sung Park, Bidipta Sarkar, Rohan Taori, Yusuke Noda, Demetri Terzopoulos, Yejin Choi, Katsushi Ikeuchi, Hoi Vo, Li Fei-Fei, and Jianfeng Gao. Agent ai: Surveying the horizons of multimodal interaction, 2024. 2

[16] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers, 2021. 3

[17] Yue Fan, Xiaojian Ma, Rujie Wu, Yuntao Du, Jiaqi Li, Zhi Gao, and Qing Li. Videoagent: A memory-augmented multimodal agent for video understanding. In *European Conference on Computer Vision*, pages 75–92. Springer, 2025. 1, 2, 3

[18] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition, 2016. 1, 2

[19] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 1, 2

[20] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023. 2

[21] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18995–19012, 2022. 5

[22] Wenyi Hong, Weihan Wang, Ming Ding, Wenmeng Yu, Qingsong Lv, Yan Wang, Yean Cheng, Shiyu Huang, Junhui Ji, Zhao Xue, Lei Zhao, Zhuoyi Yang, Xiaotao Gu, Xiaohan Zhang, Guanyu Feng, Da Yin, Zihan Wang, Ji Qi, Xixuan Song, Peng Zhang, Debing Liu, Bin Xu, Juanzi Li, Yuxiao Dong, and Jie Tang. Cogvlm2: Visual language models for image and video understanding, 2024. 3

[23] Wenyi Hong, Weihan Wang, Ming Ding, Wenmeng Yu, Qingsong Lv, Yan Wang, Yean Cheng, Shiyu Huang, Junhui Ji, Zhao Xue, et al. Cogvlm2: Visual language models for image and video understanding. *arXiv preprint arXiv:2408.16500*, 2024. 5, 6

[24] Wenyi Hong, Weihan Wang, Qingsong Lv, Jiazheng Xu, Wenmeng Yu, Junhui Ji, Yan Wang, Zihan Wang, Yuxiao Dong, Ming Ding, et al. Cogagent: A visual language model for gui agents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14281–14290, 2024. 2

[25] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Weli-

hinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 5, 6

[26] Lawrence Jang, Yinheng Li, Charles Ding, Justin Lin, Paul Pu Liang, Dan Zhao, Rogerio Bonatti, and Kazuhito Koishida. Videowebarena: Evaluating long context multimodal agents with video understanding web tasks, 2024. 2

[27] Peng Jin, Ryuichi Takanobu, Caiwan Zhang, Xiaochun Cao, and Li Yuan. Chat-univi: Unified visual representation empowers large language models with image and video understanding, 2023. 3

[28] Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Chong Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Ruslan Salakhutdinov, and Daniel Fried. Visualwebarena: Evaluating multimodal agents on realistic visual web tasks, 2024. 2

[29] Bruno Korbar, Yongqin Xian, Alessio Tonioni, Andrew Zisserman, and Federico Tombari. Text-conditioned resampler for long form video understanding. In *European Conference on Computer Vision*, pages 271–288. Springer, 2025. 1

[30] Somnath Kumar, Yash Gadhia, Tanuja Ganu, and Akshay Nambi. Mmctagent: Multi-modal critical thinking agent framework for complex visual reasoning, 2024. 3

[31] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22195–22206, 2024. 1

[32] Yunxin Li, Xinyu Chen, Baotain Hu, and Min Zhang. Llms meet long video: Advancing long video comprehension with an interactive visual adapter in llms, 2024. 1, 3

[33] 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, 2025. 1

[34] 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

[35] Ji Lin, Hongxu Yin, Wei Ping, Yao Lu, Pavlo Molchanov, Andrew Tao, Huizi Mao, Jan Kautz, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models, 2023. 2

[36] Ming C Lin and Shan Yang. Vila: Efficient video-language alignment for video question answering. 1

[37] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 2024. 5

[38] Hao Liu, Wilson Yan, Matei Zaharia, and Pieter Abbeel. World model on million-length video and language with ringattention. *arXiv preprint arXiv:2402.08268*, 2024. 1

[39] Ziyu Ma, Chenhui Gou, Hengcan Shi, Bin Sun, Shutao Li, Hamid Rezatofighi, and Jianfei Cai. Drvideo: Document retrieval based long video understanding. *arXiv preprint arXiv:2406.12846*, 2024. 2

[40] Ziyu Ma, Chenhui Gou, Hengcan Shi, Bin Sun, Shutao Li, Hamid Rezatofighi, and Jianfei Cai. Drvideo: Document retrieval based long video understanding, 2024. 3

[41] Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. *arXiv preprint arXiv:2308.09126*, 2023. 5

[42] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets: Deep networks for video classification, 2015. 2

[43] Pinelopi Papalampidi, Skanda Koppula, Shreya Pathak, Justin Chiu, Joe Heyward, Viorica Patraucean, Jiajun Shen, Antoine Miech, Andrew Zisserman, and Aida Nematzdeh. A simple recipe for contrastively pre-training video-first encoders beyond 16 frames. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14386–14397, 2024. 3

[44] Jongwoo Park, Kanchana Ranasinghe, Kumara Kahatapitiya, Wonjeong Ryoo, Donghyun Kim, and Michael S Ryoo. Too many frames, not all useful: Efficient strategies for long-form video qa. *arXiv preprint arXiv:2406.09396*, 2024. 1, 3, 5, 6

[45] Ruchit Rawal, Khalid Saifullah, Ronen Basri, David Jacobs, Gowthami Somepalli, and Tom Goldstein. Cinepile: A long video question answering dataset and benchmark. In *Synthetic Data for Computer Vision Workshop@ CVPR 2024*. 2

[46] Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. Timechat: A time-sensitive multimodal large language model for long video understanding. *arXiv preprint arXiv:2312.02051*, 2023. 3

[47] Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. Timechat: A time-sensitive multimodal large language model for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14313–14323, 2024. 5, 6

[48] Mattia Soldan, Alejandro Pardo, Juan León Alcázar, Fabian Caba, Chen Zhao, Silvio Giancola, and Bernard Ghanem. Mad: A scalable dataset for language grounding in videos from movie audio descriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5026–5035, 2022. 1

[49] Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Xun Guo, Tian Ye, Yan Lu, Jenq-Neng Hwang, et al. Moviechat: From dense token to sparse memory for long video understanding. *arXiv preprint arXiv:2307.16449*, 2023. 1, 2, 3

[50] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning, 2019. 3

[51] Yuchong Sun, Hongwei Xue, Ruihua Song, Bei Liu, Huan Yang, and Jianlong Fu. Long-form video-language pre-training with multimodal temporal contrastive learning. *Advances in neural information processing systems*, 35:38032–38045, 2022. 3

[52] Yunlong Tang, Jing Bi, Siting Xu, Luchuan Song, Susan Liang, Teng Wang, Daoan Zhang, Jie An, Jingyang Lin, Rongyi Zhu, Ali Vosoughi, Chao Huang, Zeliang Zhang, Pinxin Liu, Mingqian Feng, Feng Zheng, Jianguo Zhang,

Ping Luo, Jiebo Luo, and Chenliang Xu. Video understanding with large language models: A survey, 2024. 2

[53] Makarand Tapaswi, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. Movieqa: Understanding stories in movies through question-answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4631–4640, 2016. 1

[54] Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024. 2

[55] Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024. 1

[56] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training, 2022. 3

[57] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks, 2015. 1, 2

[58] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition, 2016. 2

[59] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 3, 5, 7

[60] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 1

[61] 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, 2024. 1, 5, 6

[62] Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. Videoagent: Long-form video understanding with large language model as agent. *European Conference on Computer Vision (ECCV)*, 2024. 1, 2, 3, 5, 6

[63] Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Jilan Xu, Zun Wang, et al. Internvideo2: Scaling video foundation models for multimodal video understanding. *arXiv preprint arXiv:2403.15377*, 2024. 1, 3

[64] Yuxuan Wang, Cihang Xie, Yang Liu, and Zilong Zheng. Videollamb: Long-context video understanding with recurrent memory bridges. *arXiv preprint arXiv:2409.01071*, 2024. 1, 3

[65] Ying WANG, Yanlai YANG, and Mengye REN. Lifelongmemory: Leveraging llms for answering queries in long-form egocentric videos. *arXiv preprint arXiv:2312.05269*, 2024. 1, 3

[66] Ziyang Wang, Shoubin Yu, Elias Stengel-Eskin, Jaehong Yoon, Feng Cheng, Gedas Bertasius, and Mohit Bansal. Videotree: Adaptive tree-based video representation for llm reasoning on long videos, 2024. 1, 2, 3, 5, 6

[67] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022. 4

[68] Yuetian Weng, Mingfei Han, Haoyu He, Xiaojun Chang, and Bohan Zhuang. Longvlm: Efficient long video understanding via large language models. In *European Conference on Computer Vision*, pages 453–470. Springer, 2025. 1

[69] Chao-Yuan Wu and Philipp Krahenbuhl. Towards long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1884–1894, 2021. 1

[70] Chao-Yuan Wu, Yanghao Li, Karttikeya Mangalam, Haoqi Fan, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Memvit: Memory-augmented multiscale vision transformer for efficient long-term video recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13587–13597, 2022. 3

[71] Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. Longvideobench: A benchmark for long-context interleaved video-language understanding. *CoRR*, 2024. 1

[72] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification, 2018. 2

[73] Tianbao Xie, Fan Zhou, Zhoujun Cheng, Peng Shi, Luoxuan Weng, Yitao Liu, Toh Jing Hua, Junning Zhao, Qian Liu, Che Liu, et al. Openagents: An open platform for language agents in the wild. In *ICLR 2024 Workshop on Large Language Model (LLM) Agents*. 2

[74] Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *arXiv preprint arXiv:2404.07972*, 2024. 2

[75] Jiaqi Xu, Cuiling Lan, Wenxuan Xie, Xuejin Chen, and Yan Lu. Retrieval-based video language model for efficient long video question answering, 2023. 1, 3

[76] Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents, 2023. 2

[77] Shoubin Yu, Jaemin Cho, Prateek Yadav, and Mohit Bansal. Self-chained image-language model for video localization and question answering. *NeurIPS*, 2023. 3

[78] Shoubin Yu, Jaemin Cho, Prateek Yadav, and Mohit Bansal. Self-chained image-language model for video localization and question answering. *Advances in Neural Information Processing Systems*, 36, 2024. 1

[79] Ce Zhang, Taixi Lu, Md Mohaiminul Islam, Ziyang Wang, Shoubin Yu, Mohit Bansal, and Gedas Bertasius. A simple llm framework for long-range video question-answering. *arXiv preprint arXiv:2312.17235*, 2023. 2

[80] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 543–553, 2023. 3

[81] Hongjie Zhang, Yi Liu, Lu Dong, Yifei Huang, Zhen-Hua Ling, Yali Wang, Limin Wang, and Yu Qiao. Movqa: A benchmark of versatile question-answering for long-form movie understanding. *arXiv preprint arXiv:2312.04817*, 2023. 2

[82] 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. 5

[83] Yuanhan Zhang, Bo Li, haotian Liu, Yong jae Lee, Liangke Gui, Di Fu, Jiashi Feng, Ziwei Liu, and Chunyuan Li. Llava-next: A strong zero-shot video understanding model, 2024. 3, 6

[84] Jun Zhao, Can Zu, Hao Xu, Yi Lu, Wei He, Yiwen Ding, Tao Gui, Qi Zhang, and Xuanjing Huang. Longagent: Scaling language models to 128k context through multi-agent collaboration. *arXiv preprint arXiv:2402.11550*, 2024. 2

[85] Long Zhao, Nitesh Bharadwaj Gundavarapu, Liangzhe Yuan, Hao Zhou, Shen Yan, Jennifer J Sun, Luke Friedman, Rui Qian, Tobias Weyand, Yue Zhao, et al. Videoprism: A foundational visual encoder for video understanding. In *Forty-first International Conference on Machine Learning*. 1

[86] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations, 2020. 3