This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

VIDEOTREE: Adaptive Tree-based Video Representation for LLM Reasoning on Long Videos

Ziyang Wang* Shoubin Yu* Elias Stengel-Eskin*
Jaehong Yoon Feng Cheng Gedas Bertasius Mohit Bansal
UNC Chapel Hill

https://videotree2024.github.io/

Abstract

Long-form video understanding is complicated by the high redundancy of video data and the abundance of query-irrelevant information. To tackle these challenges, we propose VIDEOTREE*, a training-free framework which builds a query-adaptive and hierarchical video representation for LLM reasoning over long-form videos. First,* VIDEOTREE *extracts query-relevant information from the input video through an iterative process, progressively refining the selection of keyframes based on their relevance to the query. Furthermore,* VIDEOTREE *leverages the inherent hierarchical structure of long video data, which is often overlooked by existing LLM-based methods. Specifically, we incorporate multi-granularity information into a tree-based representation, allowing* VIDEOTREE *to extract query-relevant details from long videos in a coarse-to-fine manner. This enables the model to effectively handle a wide range of video queries with varying levels of detail. Finally,* VIDEOTREE *aggregates the hierarchical query-relevant information within the tree structure and feeds it into an LLM reasoning model to answer the query. Our experiments show that our method improves both reasoning accuracy* and *efficiency. Specifically,* VIDEOTREE *outperforms existing training-free approaches on EgoSchema and NExT-QA with less inference time, achieving* 61.1% *and* 75.6% *accuracy on the test set without additional video-specific training. Moreover, on the long split of Video-MME (average 44 minutes),* VIDEOTREE *achieves better performance than GPT-4V and many other MLLMs that were extensively trained on video data.*

1. Introduction

With the surge in accessible long video content and the growing importance of applications such as long-form human behavior analysis and movie analysis, developing models capable of reasoning over and answering questions about long-form videos has become increasingly crucial. Recently, several approaches [19, 72, 89] have emerged that leverage the long-sequence reasoning capabilities of Large Language Models (LLMs) to tackle the challenge in long-form video understanding in a training-free manner. Typically, these approaches leverage vision-language models (VLM) to caption densely sampled frames, thus representing the video in text format. This text representation is then subsequently fed into an LLM, which reasons over the video and responds to the provided query. Although this strategy has demonstrated great potentials on long-form video understanding benchmarks, it still faces two major limitations:

1) Informational Overload: Long videos inherently contain high levels of information redundancy, and current approaches [7, 89] lack a principled method to effectively address this challenge. A deluge of redundant and irrelevant

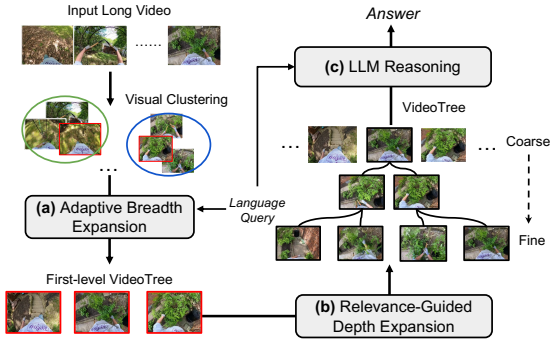

Figure 1. Overview of VIDEOTREE for LLM reasoning on long videos. Given the long video input, we first apply adaptive breadth expansion to identify the first-level keyframes for VIDEOTREE. Next, we use relevance-guided depth expansion to explore the inherent hierarchical structure of the video, forming a tree-based representation. Finally, the coarse-to-fine information extracted by VIDEOTREE is fed into the LLM reasoner to answer the query.

*Equal contribution.

information can overwhelm the LLM, leading to mistakes in long-form video reasoning and reduced efficiency.

2) Inability to Capture the Coarse-to-Fine Video Structure: Existing approaches [67, 89] often simplify video content into a list of captions without any structure, failing to account for the hierarchical nature of video information. Especially in long videos, some regions are information-dense – requiring fine-grained temporal understanding – while others are irrelevant to the query, or information-sparse. Because of this, existing approaches not only suffer from overload problems, as mentioned above, but also omit key information from the captions, leading to missed details.

These limitations underscore the pressing need for a new long-form video understanding method. To this end, we introduce **VIDEOTREE**, a training-free framework for long-form video understanding. VIDEOTREE dynamically extracts query-relevant keyframes from the video input in a coarse-to-fine manner and organizes them within a tree structure, with child nodes representing more fine-grained information. VIDEOTREE is *adaptive*, meaning that our method allocates more frames to relevant video regions and fewer frames to irrelevant ones based on the given query. VIDEOTREE is also *hierarchical*. Unlike existing approaches [67, 89], which treat video as a list of frames, we explore the inherent structure within the video data (e.g., events, scenes) to extract fine-grained information relevant to the query.

VIDEOTREE relies on three crucial steps: **adaptive breadth expansion** (Fig. 1a), **relevance-guided depth expansion** (Fig. 1b), and **LLM-based reasoning** (Fig. 1c). To address redundancy in long videos, VIDEOTREE first leverages an adaptive breadth expansion module to extract query-relevant information, forming the initial level of representation. We utilize an iterative process of visual clustering, keyframe captioning, and relevance scoring until sufficient query-relevant information is gathered. Compared to existing approaches [19, 89] that rely on dense frame captions, VIDEOTREE selects only sparse keyframes for captioning, which significantly improves inference efficiency and helps avoid irrelevant information that could interfere with accurate video reasoning. To capture more fine-grained information, we introduce a relevance-guided depth expansion step that adds finer, query-specific details in a hierarchical structure, forming a tree-based representation. Finally, we generate video descriptions from the structured representation using a captioner and provide them, along with the query, to the LLM for long video reasoning.

We demonstrate the effectiveness and efficiency of VIDEOTREE by evaluating it on two mainstream long video question answering (LVQA) datasets, EgoSchema [42]and NExT-QA [81]. Compared existing training-free approaches, VIDEOTREE achieves 2.1% and 4.3% improvements on EgoSchema(subset) and NExT-QA validation set with less inference time or LLM calls. To further validate VIDEOTREE effectiveness on very long videos, we test our method on the long split of the recent Video-MME benchmark [10] and VIDEOTREE achieves better performance than the strong proprietary GPT-4V model. Our ablation studies show that VIDEOTREE outperforms the the same category methods (VideoAgent [67] and LLoVi [89]) under all number of captions and observes better efficiency-effectiveness trade-off. We further provide addition results on open-source LLM, where VIDEOTREE shows strong generalization ability across different language backbone models and achieves 4.8% improvements against the LangRepo approach [19].

2. Related Work

Structural Video Representation. Video understanding [26, 30, 32, 34, 35, 39, 52, 56, 61, 64, 74, 76, 78, 83] has shown impressive advancement in both views of comprehension and efficiency. Recently, several video-language methods [1, 15, 28, 38, 50, 53, 77, 80, 84, 85, 87, 88] have further introduced a structured understanding of video frames to allow compact and efficient recognition of scene contexts. For example, HierVL [1] proposes a *bottom-up* hierarchical video-language embedding that capture video representations across short and long time periods. VideoReCap [15] introduces a progressive video captioning approach that generates short clip-level captions and summarizes them into longer segments. These methods process long videos by progressively building high-level knowledge from local temporal information, i.e. in a bottom-up fashion that first captures all low-level details and then aggregates. This results in significant computational and time overhead. In contrast, inspired by the existing coarse-to-fine video understanding works [73, 79], VIDEOTREE proposes a novel top-down approach with a tree structure, enabling efficient and effective long video understanding by dynamically extracting query-relevant keyframes for LLM reasoning.

Video Understanding with LLMs. Inspired by the powerful reasoning capabilities of LLMs, recent works have explored using LLMs to address complex video-related tasks. Since LLMs primarily process text, various methods [2, 12, 18, 22, 25, 27, 29, 31, 40, 44, 59, 71, 75, 90**?**] have been developed to efficiently train multimodal projectors to connect the visual encoder and LLMs or leverage caption-centric information. Past works [6, 9, 19, 21, 58, 60, 65, 67, 72] has investigated training-free combinations of captioners and LLMs for video understanding. Specifically, LLoVi [89] proposes a simple language-guided video understanding method. First, it extracts short-term video descriptions with a captioning model, and then an LLM summarizes these dense captions and responds to the given prompt. VideoAgent [67] introduces a multi-round frame search strategy using an LLM agent. Unlike existing approaches, we propose a novel

method to extract the key information from videos in an adaptive and coarse-to-fine manner with the agent, improving both efficiency and performance on long video understanding tasks. Moreover, VIDEOTREE improves interpretability by highlighting key visual clues for LLM reasoning via its human-readable tree structure.

3. VIDEOTREE Method

We present VIDEOTREE, a framework for constructing a query-adaptive, hierarchical video representation for efficient LLM reasoning over long videos. As illustrated in Fig. 2, the VIDEOTREE framework consists of three main steps: adaptive breadth expansion, relevance-guided depth expansion, and LLM video reasoning. Given the highly redundant nature of long videos, VIDEOTREE first leverages an adaptive breadth expansion module to extract query-relevant information from the video, forming the initial level of representation (Sec. 3.1). To capture finer-grained details, we propose a relevance-guided depth expansion module that progressively adds finer-grained, query-specific details to in a hierarchical manner, forming a tree-based representation (Sec. 3.2). Finally, we extract the video description from the constructed tree representation by using a captioner to caption selected frames. We feed it, along with the query, into the LLM for long video reasoning (Sec. 3.3).

3.1. Adaptive Breadth Expansion

Video data is often highly redundant, and long videos can contain substantial amounts of irrelevant information relative to the given video query. Addressing this redundancy and filtering out irrelevant content is crucial for efficient and effective long video understanding. Existing approaches [66, 86] select a fixed number of keyframes as the key information. However, as discussed in Sec. 1, this fixed keyframe selection is sub-optimal for a general long video-language understanding framework, since the information density varies across videos—some contain numerous scene changes, while others remain largely static. To address this, we propose an adaptive breadth expansion module that constructs the first level of the tree representation by dynamically identifying keyframes that are relevant to the given query. Specifically, as shown in the left of Fig. 2 (Step 1), given the video and a query about it, we build the first level of the tree by iterating three operations: **visual clustering**, **cluster captioning,** and **relevance scoring**. These operations first group similar frames together, then generate captions for each cluster, and use the LLM to determine how relevant each cluster is to the query. VIDEOTREE iterate these operations until getting enough query-relevant information from long videos in an *adaptive* manner. In the following paragraphs, we provide a detailed motivation and introduction for each operation.

Visual Clustering. To reduce the redundancy, we first propose a visual clustering operation that groups the video frames based on semantic similarity, allowing the model to focus on representative frames from each cluster while discarding repetitive or irrelevant content. Specifically, given a video sequence $V = (F_1, F_2..., F_n)$, where F_i is the frame at the time step i and n is the length of the video, we extract visual features for each frame with the pre-trained visual encoder [57] E, such that $f_i = E(F_i)$, where $f_i \in \mathbb{R}^d$ is the visual features extracted by the frame F_i. These features serve as a compact representation of each frame's visual content, capturing diverse semantics of each frame, such as scenes and objects. We then use K-Means clustering [41] to group frame features into k distinct clusters based on their similarity, which we denote as:

$$(C_1, \ldots, C_k), (c_1, \ldots, c_k) = \text{K-Means}((f_1, \ldots, f_n), k) \quad (1)$$

where, C_i is the ith cluster that groups multiple frames, c_i is the centroid vector for the ith cluster and k is the number of clusters. This clustering process reduces the redundancy within the video by converting the input from n frames into k clusters of similar frames (where $n \gg k$), effectively summarizing the video into k keyframes (cluster center frame) that capture the essential semantics.

Cluster Captioning. To better extract the key semantics from each cluster, we leverage a captioner to convert the keyframe information (a single frame or short clip around the keyframe) from each cluster to a textual description. Specifically, for the cluster C_i, we find the keyframe F_i that is closest to the centroid vector c_i and consider it as the keyframe of the ith cluster. We then feed the extracted keyframe (or the key clip) into the VLM-based captioner $Cap(\cdot)$ [36, 93] and obtain a text caption $t_i = Cap(F_i)$ for each cluster. These text captions serve as detailed descriptions of the key semantics from the corresponding clusters.

Relevance Scoring. To encourage the model to extract query-relevant information, after obtaining the cluster captions t, we leverage the reasoning capability of the LLM to assess whether the extracted information are sufficient for answering the given query. To this end, we first feed all cluster captions $\{t_i \ \forall i \in [1, \ldots, k]\}$ from the last operation and the video query Q into the LLM and output a set of relevance scores $\{r_i \ \forall i \in [1, \ldots, k]\}$ for each cluster, where r_i is the relevance of the i_{th} cluster. Specifically, to obtain each r_i, we prompt the LLM with the captions and the query, asking it to assign a relevance score to each caption, with three levels: 1 (not relevant), 2 (somewhat relevant), and 3 (highly relevant). See Tab. 16 for all detailed prompts.

Then, we adaptively extract the query-relevant information within the video by iterating the clustering, captioning,

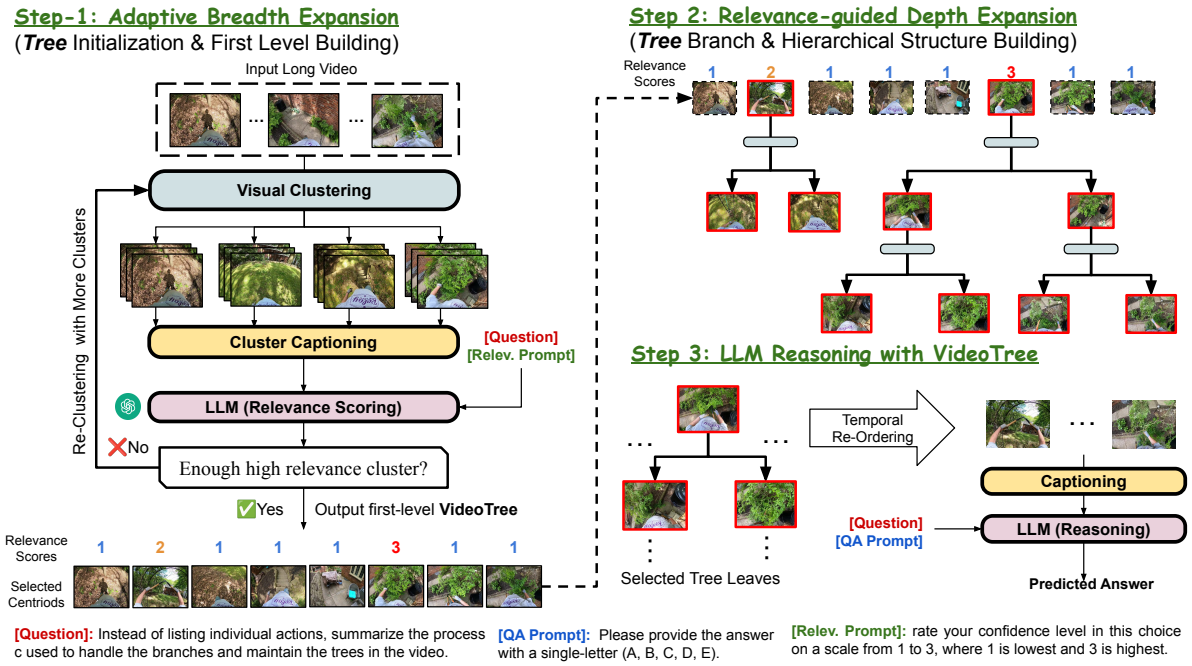

Figure 2. A detailed view of VIDEOTREE. To construct the tree structure, we begin with *Adaptive Breadth Expansion* (Step 1), which dynamically extracts query-relevant key information, considering both video and question inputs. Then, starting from the highly relevant root nodes, we explore deeper into the tree branches with *Relevance-guided Depth Expansion* (Step 2), re-clustering at each level to capture finer visual cues. Finally, we gather the selected nodes (keyframes), caption them, and arrange them in temporal order for *LLM reasoning* (Step 3).

and relevance scoring operation. Specifically, given the list of relevance scores for each cluster, we set a threshold of the number of highly relevant clusters $rele\_num\_thresh$ to decide the stop of the adaptive process. We also set a maximum value for the number of clusters ($max\_breadth$) to avoid infinite loops. If the number of highly relevant clusters is below the requirement, that indicates the information extracted from the current cluster assignment is insufficient for the LLM to answer the video query. In that case, we increase the number of clusters k by double the original number and repeat the clustering, captioning, and relevance scoring operations. If the number of high-relevance clusters meets the threshold $rele\_num\_thresh$ or the number of clusters reaches $max\_breadth$, we append the extracted clusters with their keyframes to the tree's first layer and continue to the next step (Algorithm 1, lines 2-11 for details).

3.2. Relevance-Guided Depth Expansion

After obtaining the first-level clusters and their keyframes, VIDEOTREE captures high-level query-relevant information from the video input. However, some video regions are information-dense and critical for answering the query, requiring a more detailed selection of keyframes.

Existing approaches, such as SeViLA [86] and VideoAgent [67], typically treat the selected frames as an unstructured list, overlooking the potential internal structure within the video data. To address this, as shown in Step 2 of Fig. 2, we construct a hierarchical video representation on top of the clusters from the previous breadth expansion step, allowing us to efficiently extract query-relevant details by leveraging the semantic relationships within the video data. Specifically, we expand the depth of the tree by sub-clustering the clusters with higher relevance scores from the first step. The intuition is that for high-relevance clusters, the LLM requires more detailed, granular information, while for low-relevance clusters, more information could actually lead to irrelevant details being included and could thus overwhelm the LLM, leading to incorrect reasoning.

To build the hierarchical structure, we use the relevance of a top-level cluster to determine how many levels of more granular information will be extracted from it. Since the relevance score r falls into one of three levels, we handle each first-level cluster differently based on its assigned relevance level. For "somewhat relevant" clusters, we re-cluster the first-level cluster into w sub-clusters, where w represents the tree's branch width, ensuring that more keyframes are allocated to these moderately relevant clusters. For "highly relevant" clusters, we re-cluster into a two-level tree with a branch width of w using hierarchical clustering while keeping the 1st-level cluster information from the previous K-Means step. This coarse-to-fine exploration strategy allows for the detailed extraction of relevant information, supporting comprehensive video analysis for complex

queries. We repeat this process for all first-level medium- and highly-relevant clusters and build the hierarchical structure of VIDEOTREE (lines 12-15 in Algorithm 1). After the breadth and depth expansion steps, we obtain the tree-based video representation for LLM reasoning over the long video.

3.3. LLM Video Reasoning

Finally, in order to use the LLM's ability on video reasoning, we need to present the LLM with a text-based video description. To this end, we traverse the nodes of the tree starting at the roots and expanding to the leaves, extracting keyframes from the tree's clusters at all levels and passing them into the captioner to obtain keyframe (short clip) captions. We then sort these keyframe (short clip) captions in temporal order and concatenate them into a textual description of the video. Finally, we pass this description and the input query to the LLM and output the final answer (see line 16-18 in Algorithm 1). Our full prompt is in Tab. 17.

4. Experimental Setup

Tasks & Datasets. We test VIDEOTREE on three diverse long-form video question-answering benchmarks: (1) **EgoSchema** [42], a long-range video question-answering benchmark consisting of 5K multiple choice question-answer pairs spanning 250 hours of video and covering a wide range of human activities. Our ablation studies are conducted on the official validation set of EgoSchema which contains 500 questions (referred to as the EgoSchema Subset). The videos are 180 seconds long on average. (2) **NExT-QA** [81], a video question-answering benchmark for causal and temporal reasoning. It contains 5440 videos with an average length of 44s and approximately 52K questions. NExT-QA contains 3 different question types: Temporal (Tem.), Causal (Cau.), and Descriptive (Des.). (3) **Video-MME** [10] is a recent-proposed comprehensive evaluation benchmark for video analysis. We test VIDEOTREE on the "long-term videos" split of the dataset (long split), whose average video length is 44 minutes, ranging from 30-60 minutes.

Implementation Details. We adopt GPT-4[1] [46] as our LLM for all the main results. We also provide the results with open-source LLM (Sec. 5.2) and other proprietary LLMs (Sec. 9). Following VideoAgent [67], we leverage EVA-CLIP-8B [57] as our visual encoder and also provide experimental analysis with smaller visual encoder in Sec. 5.2. Following VideoAgent [67], we leverage CogAgent [13] as the captioner for NExT-QA benchmark and use LaViLa [93] as our captioner for the EgoSchema benchmark due to its ego-centric video pretraining (we also show results in Tab. 14 using a unified captioner (LLaVA1.6-7B [36]) for all benchmarks). For Video-MME, we directly use the default unified LLaVA1.6-7B captioner. We preprocess videos by simply sampling the original frames at 1FPS for EgoSchema and NExT-QA benchmark and 0.125 FPS for Video-MME. The best-performing average number of captions for EgoSchema subset, Next-QA and Video-MME is 62.4, 12.6 and 128, respectively. We ablate our hyper-parameter choices in Sec. 9.

[1]version 1106

Evaluation Metrics. We evaluate VIDEOTREE on all datasets under the multiple-choice QA setting. We utilize standard accuracy metrics for all experiments.

5. Results

5.1. Comparison with Existing Approaches

Comparison with training-free methods. Sec. 5.1 shows a comparison of the existing training-free works and VIDEOTREE on EgoSchema and NExT-QA benchmarks. We compare our methods with three types of systems: those using all open-source LLMs [19, 51, 54], those with proprietary MLLMs [20, 49], and the most similar class to ours, which consists of methods with open-source captioners and proprietary LLMs [6, 9, 43, 65, 67, 72, 89]. Specifically, compared with the methods that leverage the same VLM (captioner) and LLM [67, 72, 89], VIDEOTREE significantly outperforms these methods on both EgoSchema and NExT-QA benchmarks. Comparing with VideoAgent [9] which also uses video-specific models (Video-LLaVA [30], ViCLIP from InternVid [69]) which were trained on extensive video data, VIDEOTREE still performs better on EgoSchema. Moreover, comparing with the methods that utilize strong multimodal LLMs, VIDEOTREE significantly outperforms IG-VLM [20] (based on GPT-4V[45]) on both EgoSchema and NExT-QA benchmarks and obtains comparable results on the EgoSchema full test set compared to the recent LVNet [49] (which uses the more powerful GPT-4o for both captioner and LLM) while outperforming LVNet on NExT-QA benchmarks. Additionally, we observe a significant gap between VIDEOTREE and the open-source LLM-based approaches, highlighting the need of strong LLM reasoning module in our method. For the sake of making a fair comparison, we also show VIDEOTREE's ability using open-source LLM in Tab. 4, where we obtain an 4.8% improvement on the EgoSchema subset. These results showcase the effectiveness of VIDEOTREE compared with existing training-free methods. Moreover, VIDEOTREE is also more efficient: we show analyses measuring the number of captions in Fig. 3 and inference time in Tab. 3, where VIDEOTREE is more efficient than relevant baselines.

[1]We de-emphasize the EgoSchema results of LangRepo since it predicts the answers via a log-likelihood classifier rather than generation, making it different from all other methods (including VIDEOTREE). We provide a comparison using the same classifier and LLM in Tab. 4 and show 4.8% improvements under same settings.

Model	(M)LLM	EgoSchema		NExT-QA			
		Sub.	Full	Tem.	Cau.	Des.	Avg.
Based on Open-source Captioners and LLMs							
MVU [51]	Mistral-13B	60.3	37.6	55.4	48.1	64.1	55.2
LangRepo [19]	Mixtral-8×7B	66.2[1]	41.2	51.4	64.4	69.1	60.9
Video-LLaVA+INTP [54]	Vicuna-7B v1.5	-	38.6	58.6	61.9	72.2	62.7
Based on Proprietary MLLMs							
IG-VLM [20]	GPT-4V	59.8	-	63.6	69.8	74.7	68.6
LVNet [49] [2]	GPT-4o	68.2	61.1	65.5	75.0	81.5	72.9
Based on Open-source Captioners and Proprietary LLMs							
ProViQ [6]	GPT-3.5	57.1	-	-	-	-	64.6
LLoVi [89]	GPT-3.5	57.6	50.3	-	-	-	-
MoReVQA [43]	PaLM-2	-	51.7	64.6	70.2	-	69.2
Vamos [65]	GPT-4	51.2	48.3	-	-	-	-
LLoVi [89]	GPT-4	61.2	-	61.0	69.5	75.6	67.7
VideoAgent [67]	GPT-4	60.2	54.1	64.5	72.7	81.1	71.3
VideoAgent [9]	GPT-4	62.8	60.2	-	-	-	-
LifelongMemory [72] [3]	GPT-4	64.1	58.6	-	-	-	-
VIDEOTREE (Ours)	GPT-4	**66.2**	**61.1**	**70.6**	**76.5**	**83.9**	**75.6**

Table 1. Comparison with other training-free methods on EgoSchema and NExT-QA. VIDEOTREE outperforms the existing approaches on all evaluation metrics.

Method	Acc
Proprietary MLLM	
GPT-4V	53.5
GPT-4o	65.3
Gemini 1.5 Pro	**67.4**
Open-Source MLLM	
LongVA	46.2
VITA	48.6
InternVL2-34B	52.6
VILA-1.5-40B	53.8
Oryx-1.5-34B	59.3
LLaVA-NeXT-Video-72B	61.5
Qwen2-VL-72B	**62.2**
Training-free Approach	
LLoVi	48.8
VIDEOTREE (Ours)	**54.2**

Table 2. Video-MME long split results. VIDEOTREE outperforms the strong proprietary GPT-4V model and many other specially-trained open-souce video MLLMs (e.g. InternVL2-34B, VILA-1.5-40B) despite being training-free.

Evaluating on Very Long Videos. To further highlight the strength of our approach on longer videos, we includue results on Video-MME [10]'s long split, which contains a diverse set of very long videos (up to 1 hour, with an average of 44 minutes). We compare our training-free method with three types of models, including proprietary MLLMs [8, 45, 47] and open-source MLLM [3, 4, 11, 37, 63, 66, 91, 92], both of which are trained on large-scale video(image) data, and training-free baseline LLoVi [89]. As shown in Sec. 5.1, compared to the training-free baseline, LLoVi, VIDEOTREE achieves a substantial 5.4% improvement on the long split of the Video-MME benchmark, demonstrating its effectiveness in understanding videos across long time-scales. Compared to proprietary MLLMs, VIDEOTREE outperforms the strong GPT-4V [45] model by 0.7%. However, there is still a gap between VIDEOTREE and powerful long-context proprietary MLLMs (GPT-4o [47], Gemini 1.5 Pro [8]). When comparing to open-source MLLMs that were extensively trained on video data, our training-free VIDEOTREE method outperforms a number of these strong MLLMs including ViLA-1.5-40B [33] and Intern-VL2 [4]. VIDEOTREE achieves strong performance without additional training on long video data.

5.2. Analysis

Below, we provide a detailed analysis of VIDEOTREE framework. All quantitative analyses are conducted on the validation subset of the EgoSchema dataset. First, we analyze the trade-off between efficiency and effectiveness, showing that our method has better efficiency *and* performance across all settings compared to existing methods. We then provide an comprehensive ablation study for different design choice of VIDEOTREE. Finally, we visualize the tree from VIDEOTREE and show the clusters VIDEOTREE chooses to expand, qualitatively supporting its quantitative gains.

5.2.1. Efficiency-Effectiveness Analysis

In Tab. 3, we show the efficiency-effectiveness trade-off of our approach compared to existing methods. Specifically, we compare VIDEOTREE with LLoVi [89] using the same GPT-4 model as LLM (and same captioner). Comparing to the best model, LLoVi, VIDEOTREE-fast (which uses fewer frames by changing the hyper-parameters) achieves a 2.4% improvement on the EgoSchema subset with only 33% the time cost. Moreover, our best model obtains a 5.0% improvement with less overall inference time compared to both LLoVi models. Profiling the inference time spent in different modules (including frame captioning, extracting keyframes/caption summarization, performing QA), we find that our hierarchical keyframe selection consumes a reasonable amount of time while significantly reducing the time cost in the captioning stage and boosting long video understanding performance. We also provide an ablation of average LLM calls and compared with VideoAgent [67] in Tab. 9 showing that VIDEOTREE requires fewer LLM calls while having better performance. These results show that

[2] For fair comparison, we de-emphasize methods that use a much stronger MLLM (GPT-4o) as both the captioner and the LLM.

[3] Reproduced results, implementation details in Sec. 11

Method	Captions	Captioner (s)	Keyfr. (s)	QA (s)	Overall (s)	Acc.
LLoVi-fast	16	2.0	0	1.9	**3.9**	57.8
LLoVi-best	180	22.4	0	2.4	24.8	61.2
VIDEOTREE-fast	**13.6**	**1.6**	4.4	**1.8**	7.8	63.6
VIDEOTREE-best	62.4	7.8	10.2	2.1	20.1	**66.2**

Table 3. Efficiency-Effectiveness comparison between LLoVi and our approach. We benchmark the time cost of VIDEOTREE and LLoVi [89], split into seconds spend in frame captioning, extracting keyframes, performing QA, and also report overall time. Using only 33% inference time, VIDEOTREE(fast) already achieves both better performance compared to LLoVi(best).

Method	# Caption	Acc.	Inf Time (s)
Based on Mistral-7B			
LLoVi	180	50.8	-
LangRepo	180	60.8	87.2
VIDEOTREE (ours)	32	**63.0**	**24.3**
Based on Mistral-8×7B (12B)			
LangRepo	180	66.2	162.1
VIDEOTREE (ours)	32	**71.0**	**50.3**

Table 4. Accuracy on the EgoSchema subset when using open-source LLM Reasoners and log-likelihood classifier. VIDEOTREE obtains better performance with less inference time on both 7B and 12B LLMs comparing to the LangRepo baseline [19].

Module	ES Acc.
VIDEOTREE	66.2
- Depth Expansion	64.4
- Adaptive Breadth Expansion	61.2

Table 5. Effect of different VIDEOTREE components. Both Adaptive Breadth Expansion and Depth Expansion modules contribute significantly to the effectiveness of VIDEOTREE.

VIDEOTREE has better effectiveness and efficiency compared to the existing method.

5.2.2. Ablation Study

In this section, we conduct ablating different parts of VIDEOTREE on the EgoSchema subset. We ablate three features: Number of captions, applying open-source LLM and different VIDEOTREE components. We include more extensive ablations (including hyper-parameters and the design of captioner/LLM/vision encoder) in Appendix Sec. 9.

Number of Captions. In Fig. 3, we compare VIDEOTREE with existing methods under different caption settings. Under similar average frame caption settings (7, 9, 11), VIDEOTREE outperforms LLoVi [89] and VideoAgent [67] by 6.5% and 2.0% on average accuracy across all three settings. Moreover, unlike the non-hierarchical VideoAgent baseline, which suffers from performance degradation after 11 frame captions (performing worse with 14 frame captions), our method continues improving, generalizing to 62.4 frame captions and achieving 6% boost at its peak. It highlights the importance of VIDEOTREE's hierarchical nature.

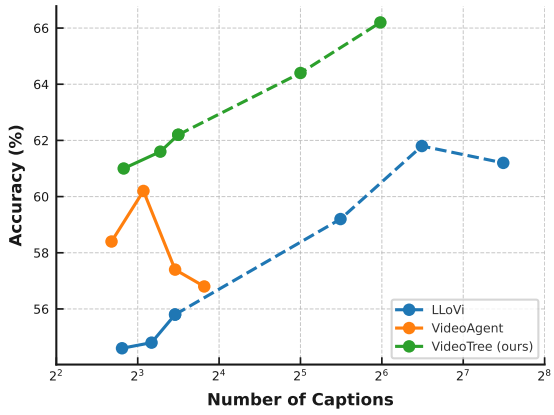

Figure 3. Ablating the number of captions. Given approximately the same number of frames, VIDEOTREE substantially outperforms LLoVi and VideoAgent. Our hierarchical nature also allows it to generalize better to more frames and perform better overall.

Open-source LLM Reasoner. To validate the effectiveness of VIDEOTREE with open-source LLM reasoners (rather than GPT4), in Tab. 4, we report the performance of VIDEOTREE using 7B and 12B versions of the Mistral model [16, 17] as the LLM reasoner. We compare with LLoVi [89] and LangRepo [19]. For a maximally fair comparison, we follow LangRepo's evaluation pipeline, using a log-likelihood classifier that scores all options and takes the highest-scoring one. VIDEOTREE substantially outperforms the baseline approaches on both 7B and 12B Mistral models while only requiring 20% of the frame captions. Specifically, compared to LangRepo, which uses complex textual summarization modules, VIDEOTREE achieves 2.2% and 4.8% better EgoSchema subset performance while using about 72.5% and 69.0% less inference time on Mistral 7B and 12B LLM, respectively. These results confirm that VIDEOTREE's effectiveness and efficiency transfer to open-source models.

VIDEOTREE Components. In Tab. 5, we report the effectiveness of the different components in VIDEOTREE. Specifically, removing the depth expansion module brings a 1.8%

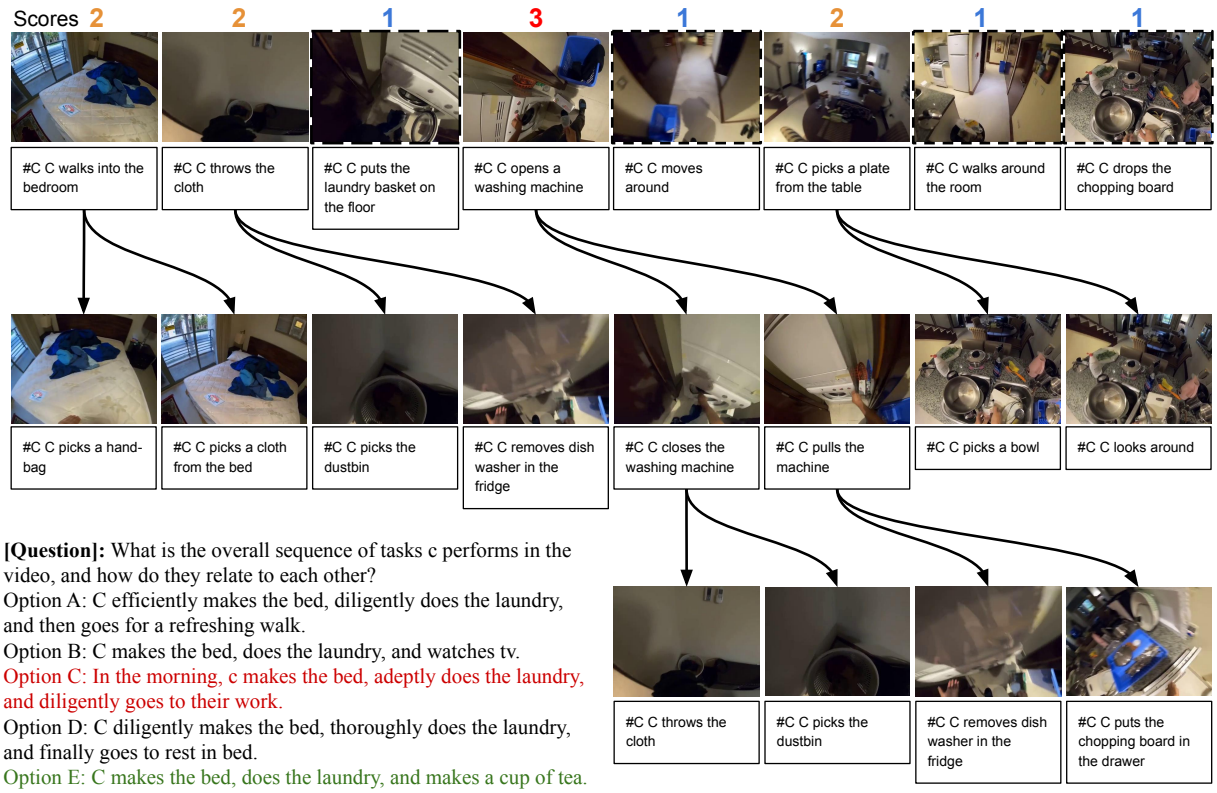

Figure 4. Qualitative examples of VIDEOTREE. Red options are answered wrongly with uniformly sampled 32 frames. Green options are answered correctly with VIDEOTREE. Best viewed in color.

drop in performance, showing the importance of the hierarchical design of VIDEOTREE. Removing the adaptive breadth expansion brings another 3.2% decrease, verifying the effectiveness of the adaptive nature of VIDEOTREE.

5.2.3. Qualitative Analysis

In Figure 4, we visualize qualitative results from VIDEOTREE. Specifically, we show the keyframes and their captions extracted by our adaptive tree representation given a video query. This example is drawn from EgoSchema, and shows the query format, which consists of a query and multiple-choice answers. With the proposed VIDEOTREE strategy, we split a complex multi-scene video (*e.g. cleaning house across rooms*) into several key scenes via visual clustering and determine the most query-relevant scene via the relevance score. We then obtain more fine-grained visual cues by descending into each relevant cluster (Levels 2 and 3 in Figure 4). For example *"C opens a washing machine"* is deemed highly relevant to the question, which asks about the sequence of events. At the same time, frames like *"C moves around"* are deemed irrelevant to the query and not expanded. In the end, VIDEOTREE shows a dynamic ability to select relevant segments and answer the given question correctly with only 50% of the baseline's 32 input captions. The LLoVi (fixed uniformly sampling) fails to correctly answer the question, sampling a large number of redundant and irrelevant frames. We also provide additional qualitative results in supplementary materials Sec. 12.

6. Conclusion

In this work, we proposed VIDEOTREE, an adaptive and hierarchical framework for LLM reasoning over long-form videos. VIDEOTREE adaptively extracts query-relevant keyframes from the video input in a coarse-to-fine manner and organizes them into a hierarchical representation, enabling the LLM to effectively handle complex queries. VIDEOTREE resulted in strong performance on three popular datasets (EgoSchema, NExT-QA, and Video-MME), while also improving efficiency by reducing the inference time and LLM calls. In our qualitative analysis, we showed that given a complex multi-scene video and its query, VIDEOTREE is capable of extracting key scenes and zooming into more detailed information that is highly related to the query. In the future, as more advanced captioners and stronger LLMs become available, the modular design of VIDEOTREE holds the potential for even greater performance and adaptability.

Acknowledgments

We thank Ce Zhang, David Wan, and Jialu Li for their helpful discussions, and the reviewers for their feedback. This work was supported by DARPA ECOLE Program No. HR00112390060, NSF-AI Engage Institute DRL-2112635, DARPA Machine Commonsense (MCS) Grant N66001-19-2-4031, ARO Award W911NF2110220, ONR Grant N00014-23-1-2356, Sony Faculty Innovation award, Laboratory for Analytic Sciences via NC State University, and Accelerate Foundation Models Research program. The views contained in this article are those of the authors and not of the funding agency.

References

[1] Kumar Ashutosh, Rohit Girdhar, Lorenzo Torresani, and Kristen Grauman. Hiervl: Learning hierarchical video-language embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23066–23078, 2023. 2

[2] Jun Chen, Deyao Zhu, Kilichbek Haydarov, Xiang Li, and Mohamed Elhoseiny. Video ChatCaptioner: Towards enriched spatiotemporal descriptions, 2023. 2

[3] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv preprint arXiv:2312.14238*, 2023. 6

[4] 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. 6

[5] Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, and Lidong Bing. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. *arXiv preprint arXiv:2406.07476*, 2024. 1

[6] Rohan Choudhury, Koichiro Niinuma, Kris M. Kitani, and Laszlo A. Jeni. Zero-shot video question answering with procedural programs. *arXiv preprint arXiv:2312.00937*, 2023. 2, 5, 6

[7] Jiwan Chung and Youngjae Yu. Long Story Short: a summarize-then-search method for long video question answering, 2023. 1

[8] Machel Reid et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *ArXiv*, abs/2403.05530, 2024. 6

[9] Yue Fan, Xiaojian Ma, Rujie Wu, Yuntao Du, Jiaqi Li, Zhi Gao, and Qing Li. Videoagent: A memory-augmented multimodal agent for video understanding. *arXiv preprint arXiv:2403.11481*, 2024. 2, 5, 6

[10] Chaoyou Fu, Yuhan Dai, Yondong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *arXiv preprint arXiv:2405.21075*, 2024. 2, 5, 6

[11] Chaoyou Fu, Haojia Lin, Zuwei Long, Yunhang Shen, Meng Zhao, Yifan Zhang, Shaoqi Dong, Xiong Wang, Di Yin, Long Ma, Xiawu Zheng, Ran He, Rongrong Ji, Yunsheng Wu, Caifeng Shan, and Xing Sun. Vita: Towards open-source interactive omni multimodal llm, 2024. 6

[12] Bo He, Hengduo Li, Young Kyun Jang, Menglin Jia, Xuefei Cao, Ashish Shah, Abhinav Shrivastava, and Ser-Nam Lim. Ma-lmm: Memory-augmented large multimodal model for long-term video understanding, 2024. 2

[13] Wenyi Hong, Weihan Wang, Qingsong Lv, Jiazheng Xu, Wenmeng Yu, Junhui Ji, Yan Wang, Zihan Wang, Yuxuan Zhang, Juanzi Li, Bin Xu, Yuxiao Dong, Ming Ding, and Jie Tang. CogAgent: A visual language model for GUI agents, 2023. 5

[14] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, 2021. If you use this software, please cite it as below. 2

[15] Md Mohaiminul Islam, Ngan Ho, Xitong Yang, Tushar Nagarajan, Lorenzo Torresani, and Gedas Bertasius. Video ReCap: Recursive captioning of hour-long videos. *arXiv preprint arXiv:2402.13250*, 2024. 2

[16] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. 7

[17] Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mixtral of experts, 2024. 7

[18] Peng Jin, Ryuichi Takanobu, Wancai Zhang, Xiaochun Cao, and Li Yuan. Chat-UniVi: Unified visual representation empowers large language models with image and video understanding, 2024. 2

[19] Kumara Kahatapitiya, Kanchana Ranasinghe, Jongwoo Park, and Michael S Ryoo. Language repository for long video understanding. *arXiv preprint arXiv:2403.14622*, 2024. 1, 2, 5, 6, 7

[20] Wonkyun Kim, Changin Choi, Wonseok Lee, and Wonjong Rhee. An image grid can be worth a video: Zero-shot video question answering using a vlm. *arXiv preprint arXiv:2403.18406*, 2024. 5, 6, 1

[21] Dohwan Ko, Ji Soo Lee, Wooyoung Kang, Byungseok Roh, and Hyunwoo J. Kim. Large language models are temporal and causal reasoners for video question answering, 2023. 2

[22] Bruno Korbar, Yongqin Xian, Alessio Tonioni, Andrew Zisserman, and Federico Tombari. Text-conditioned resampler for long form video understanding, 2024. 2

[23] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer, 2024. 1, 2

[24] Jiapeng Li, Ping Wei, Wenjuan Han, and Lifeng Fan. Intentqa: Context-aware video intent reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11963–11974, 2023. 1

[25] KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. VideoChat: Chat-centric video understanding, 2024. 2

[26] Kunchang Li, Xinhao Li, Yi Wang, Yinan He, Yali Wang, Limin Wang, and Yu Qiao. Videomamba: State space model for efficient video understanding. *arXiv preprint arXiv:2403.06977*, 2024. 2

[27] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, Limin Wang, and Yu Qiao. MVBench: A comprehensive multi-modal video understanding benchmark, 2024. 2

[28] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. HERO: Hierarchical encoder for video+language omni-representation pre-training, 2020. 2

[29] 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. 2

[30] 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, 2023. 2, 5

[31] Bin Lin, Bin Zhu, Yang Ye, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*, 2023. 2

[32] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7083–7093, 2019. 2

[33] 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. 6

[34] Kevin Lin, Faisal Ahmed, Linjie Li, Chung-Ching Lin, Ehsan Azarnasab, Zhengyuan Yang, Jianfeng Wang, Lin Liang, Zicheng Liu, Yumao Lu, Ce Liu, and Lijuan Wang. MM-VID: Advancing video understanding with GPT-4V(ision), 2023. 2

[35] Daizong Liu, Xiaoye Qu, Yinzhen Wang, Xing Di, Kai Zou, Yu Cheng, Zichuan Xu, and Pan Zhou. Unsupervised temporal video grounding with deep semantic clustering, 2022. 2

[36] 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. 3, 5

[37] Zuyan Liu, Yuhao Dong, Ziwei Liu, Winston Hu, Jiwen Lu, and Yongming Rao. Oryx mllm: On-demand spatial-temporal understanding at arbitrary resolution. *arXiv preprint arXiv:2409.12961*, 2024. 6

[38] Haoyu Lu, Mingyu Ding, Nanyi Fei, Yuqi Huo, and Zhiwu Lu. Lgdn: Language-guided denoising network for video-language modeling, 2022. 2

[39] Fan Ma, Xiaojie Jin, Heng Wang, Yuchen Xian, Jiashi Feng, and Yi Yang. Vista-LLaMA: Reliable video narrator via equal distance to visual tokens, 2023. 2

[40] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-ChatGPT: Towards detailed video understanding via large vision and language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024)*, 2024. 2

[41] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, pages 281–297. Oakland, CA, USA, 1967. 3

[42] Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 5

[43] Juhong Min, Shyamal Buch, Arsha Nagrani, Minsu Cho, and Cordelia Schmid. MoReVQA: Exploring modular reasoning models for video question answering. *arXiv preprint arXiv:2404.06511*, 2024. 5, 6

[44] Shehan Munasinghe, Rusiru Thushara, Muhammad Maaz, Hanoona Abdul Rasheed, Salman Khan, Mubarak Shah, and Fahad Khan. Pg-video-llava: Pixel grounding large video-language models. *arXiv preprint arXiv:2311.13435*, 2023. 2

[45] OpenAI. Gpt-4v(ision) system card, 2023. 5, 6

[46] OpenAI. GPT-4 technical report, 2023. 5

[47] OpenAI. GPT-4o blog, 2024. 6

[48] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024. 2

[49] 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, 2024. 5, 6

[50] Zhiwu Qing, Shiwei Zhang, Ziyuan Huang, Yi Xu, Xiang Wang, Mingqian Tang, Changxin Gao, Rong Jin, and Nong Sang. Learning from untrimmed videos: Self-supervised video representation learning with hierarchical consistency, 2022. 2

[51] Kanchana Ranasinghe, Xiang Li, Kumara Kahatapitiya, and Michael S. Ryoo. Understanding long videos in one multimodal language model pass, 2024. 5, 6

[52] Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. TimeChat: A time-sensitive multimodal large language model for long video understanding, 2024. 2

[53] Kate Sanders, Nathaniel Weir, and Benjamin Van Durme. TV-TREES: Multimodal entailment trees for neuro-symbolic video reasoning, 2024. 2

[54] Yuzhang Shang, Bingxin Xu, Weitai Kang, Mu Cai, Yuheng Li, Zehao Wen, Zhen Dong, Kurt Keutzer, Yong Jae Lee, and Yan Yan. Interpolating video-llms: Toward longer-sequence lmms in a training-free manner, 2024. 5, 6

[55] Xiaoqian Shen, Yunyang Xiong, Changsheng Zhao, Lemeng Wu, Jun Chen, Chenchen Zhu, Zechun Liu, Fanyi Xiao, Balakrishnan Varadarajan, Florian Bordes, Zhuang Liu, Hu Xu, Hyunwoo J. Kim, Bilge Soran, Raghuraman Krishnamoorthi, Mohamed Elhoseiny, and Vikas Chandra. Longvu: Spatiotemporal adaptive compression for long video-language understanding, 2024. 1

[56] Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, Yan Lu, Jenq-Neng Hwang, and Gaoang Wang. Moviechat: From dense token to sparse memory for long video understanding, 2024. 2

[57] Quan Sun, Jinsheng Wang, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, and Xinlong Wang. EVA-CLIP-18B: Scaling clip to 18 billion parameters. *arXiv preprint arXiv:2402.04252*, 2024. 3, 5, 2

[58] Dídac Surís, Sachit Menon, and Carl Vondrick. Vipergpt: Visual inference via python execution for reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11888–11898, 2023. 2

[59] Reuben Tan, Ximeng Sun, Ping Hu, Jui hsien Wang, Hanieh Deilamsalehy, Bryan A. Plummer, Bryan Russell, and Kate Saenko. Koala: Key frame-conditioned long video-LLM, 2024. 2

[60] Jue Wang, Gedas Bertasius, Du Tran, and Lorenzo Torresani. Long-short temporal contrastive learning of video transformers, 2022. 2

[61] Junke Wang, Dongdong Chen, Chong Luo, Xiyang Dai, Lu Yuan, Zuxuan Wu, and Yu-Gang Jiang. ChatVideo: A tracklet-centric multimodal and versatile video understanding system, 2023. 2

[62] Jiawei Wang, Liping Yuan, Yuchen Zhang, and Haomiao Sun. Tarsier: Recipes for training and evaluating large video description models, 2024. 1

[63] 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. 6

[64] Rui Wang, Dongdong Chen, Zuxuan Wu, Yinpeng Chen, Xiyang Dai, Mengchen Liu, Lu Yuan, and Yu-Gang Jiang. Masked video distillation: Rethinking masked feature modeling for self-supervised video representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6312–6322, 2023. 2

[65] Shijie Wang, Qi Zhao, Minh Quan Do, Nakul Agarwal, Kwonjoon Lee, and Chen Sun. Vamos: Versatile action models for video understanding, 2023. 2, 5, 6, 1

[66] Xijun Wang, Junbang Liang, Chun-Kai Wang, Kenan Deng, Yu Lou, Ming Lin, and Shan Yang. Vila: Efficient video-language alignment for video question answering, 2024. 3, 6

[67] Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. VideoAgent: Long-form video understanding with large language model as agent. *arXiv preprint arXiv:2403.10517*, 2024. 2, 4, 5, 6, 7, 3

[68] Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, et al. Internvideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*, 2022. 1

[69] Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, Conghui He, Ping Luo, Ziwei Liu, Yali Wang, Limin Wang, and Yu Qiao. Internvid: A large-scale video-text dataset for multimodal understanding and generation, 2024. 5

[70] 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

[71] Yuxuan Wang, Yueqian Wang, Pengfei Wu, Jianxin Liang, Dongyan Zhao, and Zilong Zheng. LSTP: Language-guided spatial-temporal prompt learning for long-form video-text understanding, 2024. 2

[72] Ying Wang, Yanlai Yang, and Mengye Ren. LifelongMemory: Leveraging LLMs for answering queries in long-form egocentric videos, 2024. 1, 2, 5, 6, 4

[73] Ziyang Wang, Yi-Lin Sung, Feng Cheng, Gedas Bertasius, and Mohit Bansal. Unified coarse-to-fine alignment for video-text retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2816–2827, 2023. 2

[74] Zhanyu Wang, Longyue Wang, Zhen Zhao, Minghao Wu, Chenyang Lyu, Huayang Li, Deng Cai, Luping Zhou, Shuming Shi, and Zhaopeng Tu. Gpt4Video: A unified multimodal large language model for lnstruction-followed understanding and safety-aware generation, 2023. 2

[75] Yuetian Weng, Mingfei Han, Haoyu He, Xiaojun Chang, and Bohan Zhuang. Longvlm: Efficient long video understanding via large language models, 2024. 2

[76] 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, 2022. 2

[77] Rujie Wu, Xiaojian Ma, Hai Ci, Yue Fan, Yuxuan Wang, Haozhe Zhao, Qing Li, and Yizhou Wang. Longvitu: Instruction tuning for long-form video understanding, 2025. 2

[78] Wenhao Wu. Freeva: Offline mllm as training-free video assistant, 2024. 2

[79] Zuxuan Wu, Caiming Xiong, Yu-Gang Jiang, and Larry S. Davis. Liteeval: A coarse-to-fine framework for resource efficient video recognition, 2019. 2

[80] Fanyi Xiao, Kaustav Kundu, Joseph Tighe, and Davide Modolo. Hierarchical self-supervised representation learning for movie understanding, 2022. 2

[81] Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF conference on*

computer vision and pattern recognition, pages 9777–9786, 2021. 2, 5

[82] Junbin Xiao, Pan Zhou, Tat-Seng Chua, and Shuicheng Yan. Video graph transformer for video question answering, 2022. 1

[83] Jiaqi Xu, Cuiling Lan, Wenxuan Xie, Xuejin Chen, and Yan Lu. Retrieval-based video language model for efficient long video question answering. *arXiv preprint arXiv:2312.04931*, 2023. 2

[84] Mingze Xu, Mingfei Gao, Zhe Gan, Hong-You Chen, Zhengfeng Lai, Haiming Gang, Kai Kang, and Afshin Dehghan. Slowfast-llava: A strong training-free baseline for video large language models, 2024. 2

[85] Zongxin Yang, Guikun Chen, Xiaodi Li, Wenguan Wang, and Yi Yang. DoraemonGPT: Toward understanding dynamic scenes with large language models (exemplified as a video agent), 2024. 2

[86] 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. 3, 4, 1

[87] Abhay Zala, Jaemin Cho, Satwik Kottur, Xilun Chen, Barlas Oguz, Yashar Mehdad, and Mohit Bansal. Hierarchical video-moment retrieval and step-captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23056–23065, 2023. 2

[88] Bowen Zhang, Hexiang Hu, and Fei Sha. Cross-modal and hierarchical modeling of video and text. In *Proceedings of the european conference on computer vision (ECCV)*, pages 374–390, 2018. 2

[89] 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. 1, 2, 5, 6, 7, 3

[90] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding, 2023. 2

[91] 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. 6, 1, 2

[92] 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. 6

[93] Yue Zhao, Ishan Misra, Philipp Krähenbühl, and Rohit Girdhar. Learning video representations from large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6586–6597, 2023. 3, 5