TreeReasoner: Reinforcing Tool-Augmented Tree-of-Videos Reasoning

Anonymous CVPR submission

Paper ID 3

Abstract

We present TreeReasoner, a tool-augmented, tree-structured reasoning framework that recasts long-video understanding as an active hypothesis-verification problem over a vast visual search space. By maintaining multiple parallel reasoning paths, the model systematically explores the temporal dimension and, guided by intermediate hypotheses, invokes frame-level tools such as temporal zooming, temporal jumping, and sliding to incrementally search a minimal yet sufficient chain of evidence. The entire framework is trained end-to-end with Tree-of-Tool Relative Policy Optimization (ToT-RPO) following a supervised fine-tuning warmup, achieving superior video-understanding accuracy while decoding far fewer frames than existing methods and exhibiting interpretable temporal localization and causal-verification behaviors. Experiments on six long-video reasoning benchmarks show that TreeReasoner consistently outperforms both standard IO and naive tool-calling baselines. Transferability experiments on hallucination further confirm its generalization and reduced hallucination tendencies. These experiments validate the stability and efficiency of TreeReasoner in complex temporal scenarios.

1. Introduction

Previous progress in multimodal large language models (MLLMs) [1, 13, 14, 30, 36, 37, 42, 44, 46, 57] has driven significant advancements in the ability of models to understand videos. However, long-video understanding remains an unresolved and open challenge. Unlike image understanding or short video understanding, long-video understanding requires processing visual contextual information spanning several hours or even longer. Due to constraints in computation and memory, it is impractical to conduct a comprehensive analysis of the entire visual content.

Mainstream end-to-end long-video understanding solutions typically employ uniform frame sampling at fixed intervals or with a fixed number of frames. They alleviate computational burdens through visual compression [9, 19, 21, 31] or by expanding the context length of MLLMs [55, 64], aiming to incorporate as many frames as possible for understanding. For most long-video understanding tasks, completing the task does not actually require accessing information from the entire video. Consequently, a growing body of research has begun exploring more efficient video sampling strategies [2, 43, 47, 49, 60, 61]. These approaches leverage vision-language models to select frames that are most relevant to the specific task, thereby significantly reducing the computational overhead of long-video understanding (*Fig. 1*) . Nevertheless, since frame sampling and the model's frame understanding process often cannot be optimized in an end-to-end manner, these methods, which are analogous to agent workflows, have clearly imposed limitations on performance of long-video understanding systems (*Fig. 2*) .

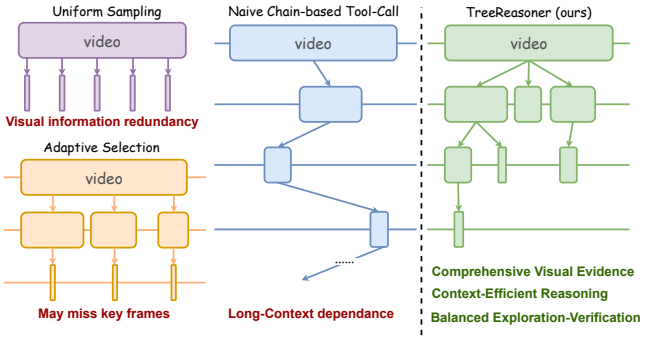
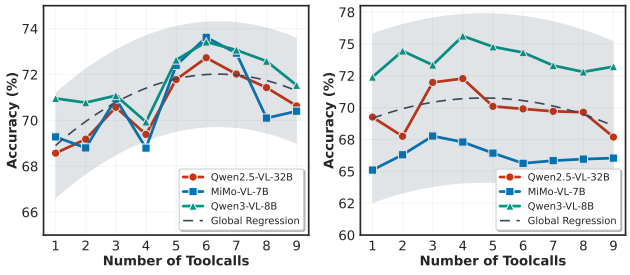


Figure 1. **Schematic illustrating different approaches for long-video understanding.** Left: Traditional sampling methods. *Uniform Sampling* (top) suffers from visual information redundancy, while *Adaptive Selection* (bottom) may miss key frames. Middle: A *Naive Chain-based Tool-Call* approach, which relies on a single reasoning path and can suffer from long-context dependence. Right (Ours): The *TreeReasoner* framework, which uses a parallel, tree-structured search to achieve comprehensive visual evidence, context-efficient reasoning, and balanced exploration-verification.

Recently, OpenAI o3 [26] has presented an alternative perspective on visual understanding and reasoning: visual perception can be treated as a tool for invocation, enabling the model to autonomously learn how to identify critical visual cues. This insight inspires us to develop an end-to-end optimization framework, which empowers the model



(a) VideoMME (b) EgoSchema

Figure 2. **Tool-call Scaling.** Empirical analysis on VideoMME (a) and EgoSchema (b) showing how performance varies under different fixed tool-call times.

to strategically explore the vast visual search space of long videos. The goal is to identify the minimal chain of evidence required to answer specific questions, thereby achieving efficient resolution of long-video understanding tasks.

In this work, we introduce TreeReasoner, a novel framework that reformulates video understanding as an active search problem through tool-augmented tree-of-thought reasoning. Our key insight is that effective video comprehension mirrors human cognitive processes: we don't passively observe entire videos but actively seek specific temporal evidence through hypothesis-driven exploration. TreeReasoner achieves this by maintaining multiple reasoning trajectories in a tree structure, where each path represents a distinct hypothesis about the temporal location and relationship of relevant events. Crucially, this tree-based exploration is intrinsically coupled with tool utilization—the model employs frame extraction, temporal navigation, and region focusing tools to gather evidence, with each tool-call guided by the current reasoning state and contributing to trajectory expansion.

The synergy between tree-structured reasoning and tool augmentation addresses the fundamental challenges of video understanding. The tree structure enables systematic exploration of multiple temporal hypotheses in parallel, preventing premature commitment to potentially incorrect temporal interpretations. Meanwhile, tool augmentation transforms each reasoning step from passive prediction to active verification, allowing the model to adaptively zoom into specific temporal regions, extract key frames based on intermediate hypotheses, and incrementally build minimal evidence chains. This bidirectional relationship—where tree exploration guides tool selection and tool outputs inform trajectory prioritization—creates an efficient search mechanism through the vast temporal space.

We train *TreeReasoner* using Tree-of-Tool Relative Policy Optimization (*ToT-RPO*), which learns to balance exploration of diverse reasoning paths with exploitation of promising trajectories. Through reinforcement learning, the model develops sophisticated temporal search strategies, learning when to extract frames, how to navigate temporal neighborhoods, and which visual details require verification. Our experiments demonstrate that TreeReasoner significantly outperforms existing methods on challenging video understanding benchmarks, achieving superior accuracy while requiring substantially fewer processed frames. The emergent search behaviors reveal interpretable reasoning patterns, including temporal bracketing, causal chain verification, and adaptive temporal resolution adjustment.

2. Related Work

2.1. Visual Agentic Reasoning

Reinforcement Learning from Verifiable Rewards (RLVR) has achieved remarkable success in the domain of Large Language Models [8, 35, 56]. Models trained with algorithms such as GRPO/PPO demonstrate strong performance across a wide range of complex reasoning and agentic tasks. Inspired by these advances, recent efforts have extended RLVR-inspired paradigms to Vision-Language Models (VLMs), yielding promising results. One line of research enhances native visual reasoning through supervised fine-tuning (SFT) or reinforcement learning (RL), encouraging models to reason truly over visual inputs rather than relying solely on the text-based reasoning capabilities inherited from their LLM backbones [16, 18, 27, 32, 41, 48, 51, 53, 54, 59, 63]. Another complementary approach equips VLMs with external tools (e.g., zoom-in/search/crop/coding api, etc) and employs end-to-end RL to further strengthen their practical agentic capabilities [17, 22, 24, 38, 39, 65, 67]. In contrast to these works, this paper focuses on naive long-video agentic reasoning and proposes a tree-based reasoning framework that generalizes beyond simple chain-of-thought (CoT) processes.

2.2. Long-Video Understanding and Reasoning

Existing end-to-end long-video Multimodal Large Language Models (MLLMs) primarily fall into two categories. First, given the visual redundancy inherent in long videos, numerous studies [9, 19, 21, 31, 46, 57] have attempted to design visual compression algorithms to reduce the number of video tokens, thereby enabling long-video understanding with acceptable computational overhead. The second research direction [13, 55, 64] draws on techniques from long-context large language models, leveraging context extension to increase the input sequence length. Additionally, a substantial body of work has focused on developing complex Agent systems for long-video understanding [2, 43, 47, 49, 60, 61], their core design involves algorithms that select key frames from long videos for understanding and reasoning. However, due to the difficulty of implementing end-to-end optimization, such methods struggle to further push the boundaries of performance. Recently, there have been many efforts [3, 5, 7, 10, 12, 20, 40] to enhance video reasoning capabilities using RL, yet these

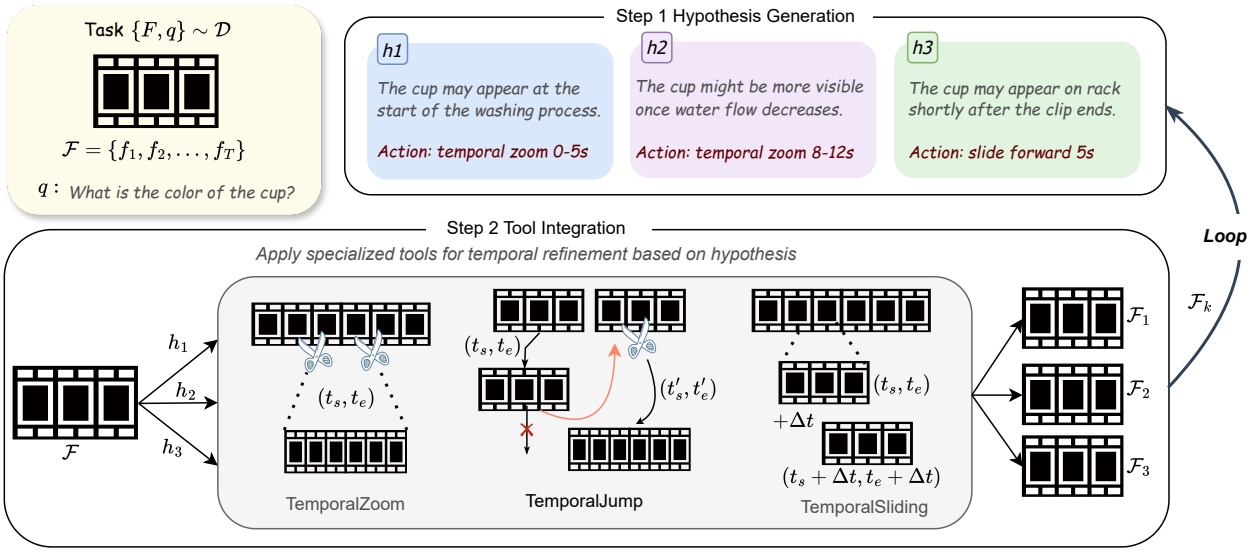


Figure 3. **Overview of TreeReasoner.** Given a question and a long video, the model first generates n parallel hypotheses about where and when relevant evidence might occur. For each hypothesis it invokes a specialized, frame-level tool—zoom, jump, or slide—to extract a short, task-specific clip. Every tool-call spawns a new child node, and the process is repeated breadth-first until an answer is reached or the budget is exhausted, yielding a minimal yet sufficient chain of evidence.

still fail to address the issue of excessive overhead during long-video understanding and reasoning. Unlike previous approaches, this paper proposes a solution that supports end-to-end optimization and enables efficient long-video understanding through temporal processing.

3. Methodology

In this section, we introduce our tool-augmented video reasoning as an active search and critic over temporal hypotheses under partial observability with tree-based expansion policy. Specifically, in section 3.1, we formulate our theoretical framework on learning objectives and constraints. Then in section 3.2, we dive into details of our reasoning process end-to-end. Finally, in section 3.3 we introduce the reward formulation and training strategy based on our proposed Tree-of-Tool Relative Policy Optimization (ToT-RPO) algorithm for tree-based video reasoning.

3.1. Problem Formulation

We formulate video reasoning process as an active search and criticism over temporal hypotheses under partial observability via the guidance of a tree-based policy. Given a video sequence $\mathcal{F} = \{f_1, f_2, \ldots, f_T\}$ with T total frames and a language query q, our reasoning objective is to discover or search an optimal evidence set $\mathcal{E}^* \subseteq \mathcal{F}$ that maximizes accuracy while minimizing computational cost. This constitutes a multi-objective optimization problem:

$$\mathcal{E}^*, \tau^* = \arg\max_{\mathcal{E}, \tau} \mathbb{P}(y^* \mid q, \mathcal{E}, \tau) \quad \textbf{s.t.} \quad C(\mathcal{E}, \tau) \leq C_{\text{budget}}.$$

Here y^* denotes the ground truth answer, τ represents the reasoning trajectory, $C(\mathcal{E}, \tau)$ measures computational cost, which contains both the number of reasoning trajectories and the computational complexity of each trajectory. C_{budget} means the controlled reasoning budget. The fundamental challenge emerges from the exponential search space $|\mathcal{P}(\mathcal{F})| = 2^T$ over the frame sequences and the partial observability constraint that limits frame access at any given time step. We model partial observability through a state-dependent visibility function $\mathcal{V}_t : \mathcal{S} \times \mathcal{A} \to 2^T$ that determines which video segments become observable after taking action a in state s. This formulation captures the realistic constraint that video understanding models cannot process entire video frames simultaneously due to memory and computational limitations. Additionally, we impose temporal coherence constraints on searched evidence frame chains through conditional dependencies:

$$\begin{aligned} &\forall \mathcal{E} = \{f_{i_1}, \ldots, f_{i_k}\}, i_1 < \cdots < i_k : \\ &\mathbb{P}(f_{i_{j+1}} | f_{i_1}, \ldots, f_{i_j}, q) > \mathbb{P}(f_{i_{j+1}} | q), \end{aligned} \tag{1}$$

thus ensuring that evidence frames form meaningful temporal narratives rather than disconnected visual elements.

3.2. Tree-of-video Reasoning

Our approach models the reasoning process as a directed tree $\mathcal{T} = (\mathcal{N}, \mathcal{E}_{\text{tree}})$ where nodes represent reasoning states and edges encode tool-augmented state transitions. Given the challenge of searching through massive frame sequences in long-video understanding, chain-like reasoning methods (e.g., Chain-of-Thought) may fail to accurately localize relevant clips from the outset. Their depth-first expansion tends to trap the policy model in irrelevant frame regions, with limited ability to recover or escape. Moreover,

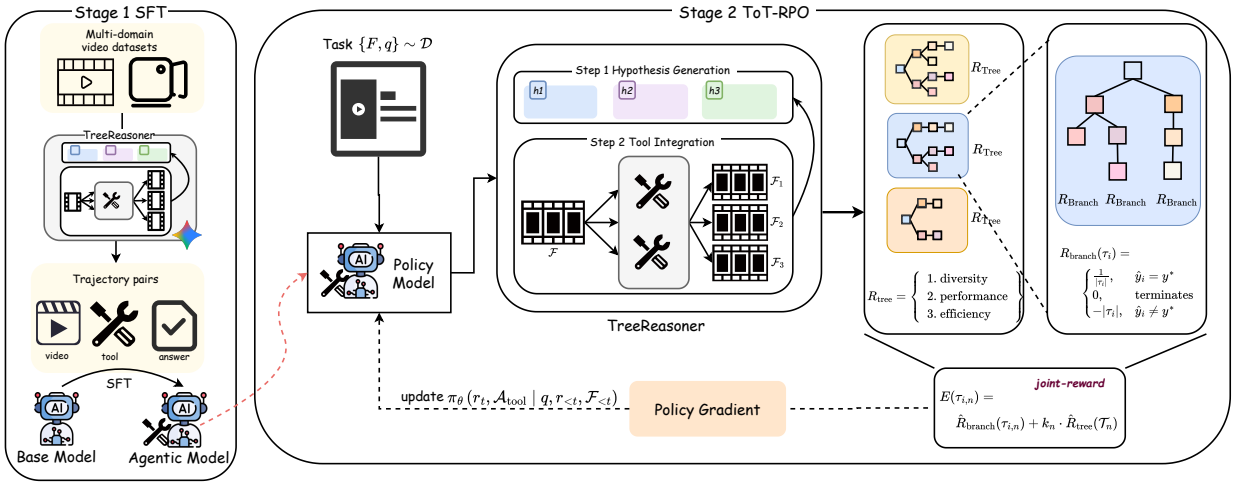


Figure 4. **Training paradigm of TreeReasoner.** *Stage 1*: the agent is warmed up with SFT on multi-turn, tool-augmented reasoning trajectories distilled from a teacher model. *Stage 2*: we continue with *ToT-RPO*, sampling entire reasoning trees per video–question pair and optimizing a composite reward that balances answer accuracy, search efficiency, tool utility, and inter-branch diversity. The whole pipeline is end-to-end and produces a single policy that jointly learns when to reason, which tool to call, and where to look next.

current RL algorithms face significant convergence difficulties in multi-turn tool-use settings. To address these issues, we adopt a breadth-first tree search strategy, enabling parallel exploration of multiple key frame intervals while progressively refining clip localization via tool interaction. This leads to earlier correct answer retrieval or timely early stopping, substantially improving reasoning efficiency.

3.2.1. Reasoning Process Representation

In our work, the reasoning state$s_t \in \mathcal{S}$ of each node $n_t \in \mathcal{N}$ is comprehensively characterized by a multi-modal tuple:

$$s_t = \langle q, \mathcal{H}_t, \mathcal{O}_t, \mathcal{R}_t, \mathcal{A}_t \rangle, \tag{2}$$

where q represents the input query. $\mathcal{H}_t = \{h_1, \ldots, h_t\}$ denotes the sequence of hidden representations encoding temporal hypotheses. $\mathcal{O}_t = \{(o_i, t_i, b_i)\}$ contains observed visual elements o_i with their temporal locations t_i and spatial bounding boxes b_i. $\mathcal{R}_t$ maintains the textual reasoning trace. $\mathcal{A}_t = \{a_1, \ldots, a_{t-1}\}$ represents the history of actions taken up to the current state.

The state transition function $\delta : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ governs how states evolve through actions. For tool-based actions, the transition incorporates environmental feedback:

$$s_{t+1} = \delta(s_t, a_t) = \begin{cases} \delta_{\text{tool}}(s_t, a_t, \text{Env}(a_t, s_t)) & \text{if } a_t \in \mathcal{A}_{\text{tool}} \\ \delta_{\text{answer}}(s_t, a_t) & \text{if } a_t \in \mathcal{A}_{\text{answer}} \\ \delta_{\text{terminal}}(s_t, a_t) & \text{if } a_t \in \mathcal{A}_{\text{terminal}} \end{cases} \tag{3}$$

where $\text{Env}(a_t, s_t)$ executes the tool action with parameters $\boldsymbol{\theta}_a$ in the video environment and returns observations.

3.2.2. Tool Integration

The action space $\mathcal{A} = \mathcal{A}_{\text{tool}} \cup \mathcal{A}_{\text{answer}} \cup \mathcal{A}_{\text{terminal}}$ encompasses three distinct categories of operations. Answer actions $\mathcal{A}_{\text{answer}}$ involve generating the final response directly to the query based on the reasoning trace and collected visual evidence. Tool actions $\mathcal{A}_{\text{tool}}$ enable active video exploration with potential environmental interaction through parameterized functions. Inspired by natural human behaviors when watching videos, we design the following concrete tools:

- **TemporalZoom**(t_s, t_e) performs temporal zooming within the current clip, extracting frames at higher temporal resolution r from time interval $[t_s, t_e]$
- **TemporalJump**(t'_s, t'_e) jumps out of the current clip to extract frames from a different temporal interval $[t'_s, t'_e]$ in the global video, avoiding trapping in a local loop.
- **Sliding**$(t_s, t_e, \Delta t)$ slides a temporal window starting from interval $[t_s, t_e]$ with stride Δt, progressively exploring adjacent temporal segments.

These tool actions are practically useful as they mirror natural human video-watching behaviors: carefully examining details within interesting segments (zooming), jumping to other parts of the video to seek relevant evidence (jumping), and smoothly browsing continuous video content (sliding). Each tool action is parameterized by a continuous parameter vector $\boldsymbol{\theta}_a \in \mathbb{R}^{d_a}$ learned through the policy network. This parameterization allows the policy model to learn optimal tool usage strategies through RL rather than relying on hand-crafted heuristics.

3.2.3. Hierarchical Tree Expansion

Tree expansion follows a principled hierarchical search strategy that balances exploration breadth with computational efficiency. At each state s_t, the policy network π_θ generates a probability distribution $\pi_\theta(a_t|s_t)$ over actions conditioned on the current state, where the policy network π_θ employs a transformer architecture with specialized attention mechanisms for processing temporal hypotheses, vi-

sual observations, and textual reasoning traces. First from a high-level perspective, the search process maintains k parallel trajectories with diversity regularization. For each trajectory i, we sample actions at timestep t according to the constrained policy distribution:

$$a_t^{(i)} \sim \pi_\theta(\cdot|s_t^{(i)}) \quad \text{s.t.} \quad \sum_{j=1}^{k} \mathrm{KL}[\pi(\cdot|s_t^{(i)}) \| \pi(\cdot|s_t^{(j)})] \geq \beta_{\text{div}}, \tag{4}$$

where the diversity constraint β_{div} prevents trajectory collapse and encourages exploration of alternative reasoning paths. This constraint is enforced through a diversity-augmented sampling procedure that rejects actions leading to excessive similarity with existing trajectories.

Then, dive into the internal trajectory, the hierarchical expansion strategy operates across multiple temporal scales through a coarse-to-fine refinement process. Initially, the search explores broad temporal regions using low temporal resolution sampling: $\mathcal{L}_{\text{coarse}}(t_s, t_e) = \{t_s + k \cdot \Delta_{\text{coarse}} : k \in \mathbb{Z}, t_s \leq t_s + k \cdot \Delta_{\text{coarse}} \leq t_e\}$ where Δ_{coarse} represents the coarse temporal resolution. Promising regions identified during coarse search are subsequently refined using higher temporal resolution: $\mathcal{L}_{\text{fine}}(t_s, t_e) = \{t_s + k \cdot \Delta_{\text{fine}} : k \in \mathbb{Z}, t_s \leq t_s + k \cdot \Delta_{\text{fine}} \leq t_e\}$ with $\Delta_{\text{fine}} \ll \Delta_{\text{coarse}}$. Note that this tree-structured, frame-granular enhancement expansion strategy can enable the policy model to concurrently focus on key frame intervals, allowing it to rapidly identify important clues or terminate search early—avoiding wasted computation on irrelevant frames or iterative trial-and-error over incorrect frame parts (as in deep tool-call chain), thereby significantly reducing inference time.

3.3. Learn to Tree-of-video Reasoning

3.3.1. Warmup SFT

To enable model to learn how to reason with tool use based on external execution feedback, we first generate multi-round multimodal reasoning data using Gemini2.5-Pro API within our agent data workflow. For an input video query q, the Gemini2.5 model generates l rounds of reasoning output R_{query}, we filter the output with wrong answers and finally construct a SFT dataset $\mathcal{D}$. This dataset is used to train the model to predict correct tool-use action $(r_t, \mathcal{A}_{\text{tool}})$ on input query q, previously selected video frame sequence $\mathcal{F}_{<t}$ and reasoning result $r_{<t}$. The SFT objective is to unlock the policy's reasoning ability with specific tool use. Formally, we minimize following negative log-likelihood loss:

$$\mathcal{L}_{\text{SFT}} = -\mathbb{E}_{(q,r,\mathcal{A}_{\text{tool}},\mathcal{F})\sim\mathcal{D}} \left[\sum_{t=1}^{T} \log \pi_\theta \left(r_t, \mathcal{A}_{\text{tool}} \mid q, r_{<t}, \mathcal{F}_{<t} \right) \right]. \tag{5}$$

3.3.2. Reinforcement Learning Via ToT-RPO

Although SFT enables the model to imitate tool-augmented reasoning trajectories, it cannot optimize long-term decision outcomes or balance exploration and efficiency. To address these limitations, we further refine TreeReasoner with *Tree-of-Tool Relative Policy Optimization (*ToT-RPO*)* [8], which extends GRPO to structured reasoning trees for end-to-end optimization over multi-branch trajectories. After the SFT stage, the model is trained via *ToT-RPO* by sampling a group of tree-based trajectories for each video–query pair (Sec 3.2) and optimizing composite rewards that integrate accuracy, reasoning efficiency, tool utility, and exploration diversity, enabling adaptive and cost-aware reasoning across tree-structured trajectories.

Algorithm 1 *ToT-RPO* Rollout

Require: An array of video-query pairs $Q = \{(q_1, \mathcal{F}_{1,0}), (q_2, \mathcal{F}_{2,0}), \ldots, (q_n, \mathcal{F}_{n,0})\}$
Ensure: Rollout responses T of tree for all $(q, v) \in Q$.

```
P ← Q                                          ▷ Init root node of tree
while P ≠ ∅ do
    S ← INFERENCE(P)                           ▷ Reason with action
    p^last ← P
    P ← ∅                                      ▷ Clean up the node queue
    for s_k in S do                            ▷ Recongize node states
        if a_k ∈ A_terminal or a_k ∈ A_answer then
            T ← T ⊕ {p_k^last ⊕ s_k}           ▷ Build tree
        else
            for a_{k,l} in s_k do              ▷ Expand child node
                s'_{k,l} ← EXECUTE(a_{k,l}, s_k)   ▷ Execute tool
                P ← P ∪ {s'_{k,l}}
            end for
        end if
    end for
end while
return T                                       ▷ Return the final responses of tree
```

Tree-structured Rollout Process. In TreeReasoner, each reasoning episode can be viewed as a stochastic tree-search process, where the policy π_θ governs both reasoning and tool-usage actions across the tree nodes. Given a video–query pair (v, q), the model generates a reasoning tree $\mathcal{T} = \{\tau_1, \tau_2, \ldots, \tau_{|\mathcal{T}|}\}$ consisting of $|\mathcal{T}|$ branch trajectories $\tau_i = \{(s_t, a_t)\}_{t=1}^{T_i}$, each representing a distinct hypothesis exploration path ending with either an answer or early termination signal. Unlike linear chain rollouts, *ToT-RPO* treats these trajectories collectively during optimization, leveraging their structural interdependence. To operationalize this process, the reasoning tree is constructed through breadth-first expansion: starting at the root node N_0, a queue P is maintained to manage nodes at each depth level, and for every node in P, the model proactively generates reasoning analysis along with an associated action. These actions encompass tool-calling actions,

Table 1. **Performance of our TreeReasoner on video understanding benchmarks.** We compare our **TreeReasoner** against baseline models (IO), chain-based **Tool-call** method, and other SOTA models on 6 datasets. The 🔥 denotes models trained with both SFT and RL.

Model	Method	VideoMME	LongVideoBench	EgoSchema	MLVU	VSIBench	TOMATO	Average
OpenAI GPT-4o [15]	IO	71.9	66.7	72.2	64.6	34.0	37.7	57.9
Gemini2.5-Flash-Lite [6]	IO	65.0	60.2	–	69.3	27.0	–	–
Qwen2.5-VL-7B [1]	IO	65.1	56.0	65.0	70.2	34.2	27.6	53.0
Qwen2.5-VL-72B [1]	IO	73.3	60.7	76.2	74.6	37.2	31.2	58.9
Llama3.2-11B-Inst[25]	IO	46.0	45.5	54.3	44.4	20.6	21.5	38.7
Gemma3-12B-IT[34]	IO	58.2	51.5	56.9	52.3	32.4	28.6	46.7
MiMo-VL 7B [33]	IO	70.3	60.4	67.4	71.3	40.8	36.1	57.7
	Tool-call	72.3	63.1	69.2	74.1	45.1	38.4	60.4
	Tool-call 🔥	75.1	65.2	71.0	78.2	47.8	40.3	62.9
	TreeReasoner	73.2	63.3	69.6	77.1	46.7	38.5	61.4
	TreeReasoner 🔥	75.9	67.8	73.7	81.8	49.6	41.5	65.0
Qwen3-VL 8B [56]	IO	71.4	58.4	73.2	78.1	59.4	34.9	62.5
	Tool-call	73.3	61.5	75.6	80.8	61.8	37.3	65.1
	Tool-call 🔥	76.2	64.8	77.8	84.3	63.1	40.5	67.8
	TreeReasoner	73.7	62.2	76.4	81.5	62.6	37.8	65.7
	TreeReasoner 🔥	77.2	65.9	80.0	85.5	64.6	41.5	69.1
Qwen2.5-VL 32B [1]	IO	70.4	57.8	69.2	71.3	37.1	29.8	55.9
	Tool-call	72.3	59.1	71.8	75.4	43.7	31.5	59.0
	Tool-call 🔥	74.8	62.1	74.6	79.8	46.1	35.2	62.1
	TreeReasoner	72.5	60.2	73.0	76.4	44.7	33.2	60.0
	TreeReasoner 🔥	76.4	62.9	77.6	80.4	47.6	37.5	63.7

answer-generation actions, and early-stopping actions (see Sec. 3.2.1). When a tool-calling action is correctly produced, the corresponding tool is executed and its feedback is appended to the next-layer nodes, which are then added to P; otherwise, that branch halts. Each node can expand into at most W child nodes, and the overall depth of the tree is capped at D, ensuring a controlled yet expressive exploratory reasoning structure.

Hierarchical Reward Design. During *ToT-RPO* training, the rollout process naturally produces a structured reasoning tree rather than a single linear trajectory. This structural property prevents direct application of conventional trajectory-level policy gradient updates. To address this, we employ a hierarchical reward formulation that evaluates both local branch trajectories and the global reasoning tree, enabling stable and efficient optimization.

At the branch-trajectory level, we define a reward function that jointly captures prediction correctness and search efficiency:

$$\mathcal{R}_{\text{branch}}(\tau_i) = \begin{cases} \frac{1}{|\tau_i|}, & \text{if } \hat{y}_i = y^*, \\ 0, & \text{if the branch terminates,} \\ -|\tau_i|, & \text{if } \hat{y}_i \neq y^*, \end{cases} \tag{6}$$

where $|\tau_i|$ denotes the number of nodes in the branch trajectory and $\hat{y}_i$ is the predicted answer. This formulation encourages the discovery of correct answers with minimal reasoning depth while heavily penalizing long yet incorrect exploration paths.

Beyond branch evaluation, we introduce a tree-level reward to assess the overall quality of the reasoning tree $\mathcal{T}_j$. This reward integrates three complementary aspects: (1) *diversity*, encouraging exploration of distinct reasoning hypotheses through average branch dissimilarity; (2) *performance*, measured by the success rate of branch trajectories arriving at correct answers; and (3) *efficiency*, reflected by the depth of the reasoning tree. Together, these components ensure that the model learns to balance explorative breadth and computational economy.

To stabilize optimization, both branch-trajectory rewards and tree-level rewards are standardized across the batch. The combined advantage $E(\tau_{i,j})$ is defined as

$$E(\tau_{i,j}) = \hat{R}_{\text{branch}}(\tau_{i,j}) + k_j \cdot \hat{R}_{\text{tree}}(\mathcal{T}_j), \tag{7}$$

where

$$\begin{cases} \hat{\mathcal{R}}_{\text{branch}}(\tau_{i,j}) = \dfrac{\mathcal{R}_{\text{branch}}(\tau_{i,j}) - \mu(\mathcal{R}_{\text{branch}}(\cdot,j))}{\sigma(\mathcal{R}_{\text{branch}}(\cdot,j))}, \\ \hat{\mathcal{R}}_{\text{tree}}(\mathcal{T}_j) = \dfrac{\mathcal{R}_{\text{tree}}(\mathcal{T}_j) - \mu(\mathcal{R}_{\text{tree}})}{\sigma(\mathcal{R}_{\text{tree}})}. \end{cases} \tag{8}$$

Here, $\mu(\cdot)$ and $\sigma(\cdot)$ denote the empirical mean and standard deviation, respectively, computed over the collection of values indicated within the parentheses.

The optimization objective follows the standard GRPO framework:

$$\mathcal{L}_{\text{ToT-RPO}} = -\mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{k=1}^{K} A_{\text{group}}(\tau)\,\log\pi_\theta(a_k|s_k) - \lambda_{\text{KL}}\,D_{\text{KL}}(\pi_\theta\|\pi_{\text{ref}})\right]. \tag{9}$$

The group-normalized advantage is computed as

$$A_{\text{group}}(\tau) = E(\tau)\cdot\text{clip}\left(\frac{\pi_\theta(\tau)}{\pi_{\text{old}}(\tau)},\,1-\delta,\,1+\delta\right) \tag{10}$$

where the clipping operation prevents excessively large updates and contributes to training robustness.

4. Experiments

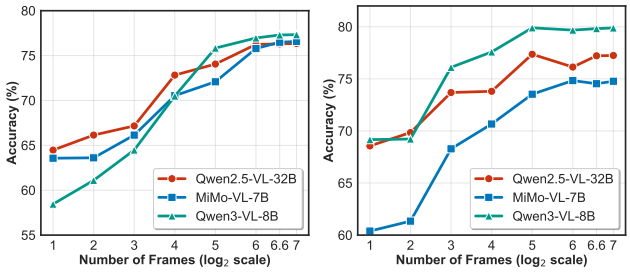
4.1. Settings

Models and baselines. We select three representative families of SOTA open-source VLMs, including Qwen2.5-VL [1], Qwen3-VL [56], and MiMo-VL [33] to evaluate our TreeReasoner framework. For each selected model family, we first assess its baseline performance, then enhance its reasoning capabilities through both standard chain-based tool-calling and our proposed TreeReasoner paradigm, with and without SFT and end-to-end RL. Additionally, we include several strong closed-source models and other competitive open-source models in our comparison directly to comprehensively demonstrate the effectiveness and robustness of our approach.

Benchmarks. We evaluate on several challenging video understanding and reasoning benchmarks, including one general video understanding task (VideoMME [11]), three long-video understanding tasks (MLVU [68], LongVideoBench [50], EgoSchema [23]) and two complex video reasoning tasks (TOMATO [29], VSIBench [58]).

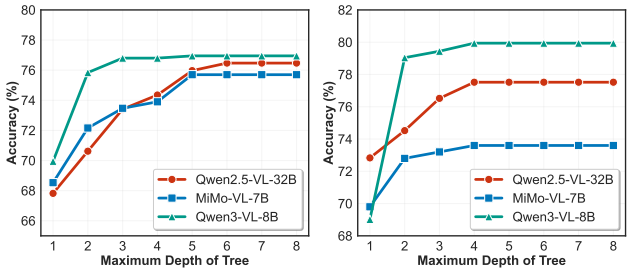
4.2. Performances of TreeReasoner

Table 1 shows the main performance of different baselines and our proposed TreeReasoner methods. Compared with the IO and naive Tool-Call baselines, TreeReasoner delivers consistent and substantial performance improvements across all six datasets. This demonstrates that TreeReasoner's parallel hypothesis exploration and early evidence verification provide immediate benefits over linear Tool-call or single-pass inference. When equipped with SFT and RL training, the TreeReasoner framework yields additional performance gains and enables multiple backbone models to reach state-of-the-art results across several benchmarks. Case study for interpretability of TreeReasoner can be found in the appendix.



(a) VideoMME (b) EgoSchema

Figure 5. **Ablation of Frame Numbers.** (a) Accuracy with various Frame Numbers on VideoMME . (b) Accuracy with various Frame Numbers on EgoSchema .



(a) VideoMME (b) EgoSchema

Figure 6. **Ablation of Max Tree Depth of TreeReasoner.** (a) Accuracy with various tree depth budgets on VideoMME. (b) Accuracy with various tree depth budgets on EgoSchema.

4.3. Analysis

Ablation on training stages. Both SFT and RL are compatible with the proposed TreeReasoner framework. To elucidate the impact of different training stages on final performance, we conducted an ablation study comparing various training configurations of TreeReasoner. As shown in Table 2, we can find that incorporating the RL stage consistently improves performance across various base models and benchmarks. These results indicate that TreeReasoner's RL process effectively refines temporal reasoning and cross-frame consistency, providing stable performance boosts regardless of model scale.

Ablation on limit of exploring depth and frame numbers for each node. Tree depth and the number of frames per node are two critical hyperparameters in our TreeReasoner reasoning framework, playing a pivotal role in balancing inference accuracy and computational efficiency. We conduct comprehensive ablation studies on these key parameters to identify the optimal configuration and to demonstrate their adaptability and robustness across diverse tasks. As shown in Fig. 5, increasing the number of sampled frames per node improves performance until saturation. Similarly, Fig. 6 shows that increasing the maximum tree depth yields consistent gains before convergence. These results indi-

Table 2. **Ablation Study on Training Stages with Different Base Models.** Accuracy(%) comparison across six video benchmarks for comparison with different base models at both SFT stage and the combined SFT with RL (SFT+RL) stage.

Base Model	Stage	VideoMME	LongVideoBench	EgoSchema	MLVU	VSIBench	TOMATO	Average
MiMo-VL 7B	SFT	73.9	65.6	71.6	79.7	47.6	39.6	63.0
	SFT+RL	75.9	67.8	73.7	81.8	49.6	41.5	65.0
Qwen3-VL 8B	SFT	75.2	63.9	78.2	83.4	63.2	39.4	67.3
	SFT+RL	77.2	65.9	80.0	85.5	64.6	41.5	69.1
Qwen2.5-VL 32B	SFT	74.4	61.0	75.4	78.4	45.7	35.6	61.8
	SFT+RL	76.4	62.9	77.6	80.4	47.6	37.5	63.7

Table 3. **Hallucination.** Accuracy(%) on Hallucer benchmark.

Model	Method	Basic	Hallucinated	Overall
MiMo-VL 7B	IO	82.8	66.8	57.6
	Tool-call	84.0	67.6	58.6
	TreeReasoner	84.3	69.7	60.7
	Tool-call 🔥	87.7	72.6	65.2
	TreeReasoner 🔥	89.9	75.9	69.1
Qwen3-VL 8B	IO	79.4	78.2	61.8
	Tool-call	81.1	81.2	64.6
	TreeReasoner	83.5	83.0	66.5
	Tool-call 🔥	84.5	85.1	69.8
	TreeReasoner 🔥	86.0	87.1	72.9
Qwen2.5-VL 32B	IO	75.2	80.4	62.0
	Tool-call	76.2	81.8	62.8
	TreeReasoner	77.9	82.3	64.6
	Tool-call 🔥	79.4	87.0	67.5
	TreeReasoner 🔥	82.6	88.7	71.8

cate that enriching frame-level visual diversity enables more comprehensive evidence aggregation and strengthens the model's structured visual reasoning process, while redundant frames offer limited additional benefit.

Transferability on Hallucination. A key challenge in the reasoning process of MLLM lies in their propensity to hallucinate—particularly when reasoning chains become overly long or complex. As attention distributions become sparse, the model may drift away from visual grounding, resulting in unfaithful or fabricated content. To assess the robustness and transferability of our reasoning framework under such conditions, we evaluate models on the Hallucer benchmark [45]. Each sample in Hallucer consists of a Basic question (testing factual understanding) and a Hallucinated counterpart (containing unverifiable or false premises). A point is awarded only when the model correctly answers both questions in a pair, ensuring that robustness against hallucination is jointly assessed with factual consistency. Experimental results show that our proposed TreeReasoner framework achieves substantially lower hallucination rates compared to standard IO and naive Tool-call, demonstrating its stronger transferability and robustness in mitigating hallucination. As shown in Table 3, TreeReasoner consistently mitigates hallucination across various backbone models and configurations.

Table 4. **Efficiency.** Efficiency analysis on LongVideoBench.

Method	Output/Input tokens	Accuracy
IO (best of 6)	9.1k / 21.1k	63.5
Tool call (best of 6) 🔥	13.2k / 49k	67.4
TreeReasoner 🔥	13.6k / 36.2k	67.8

Efficiency. To evaluate the computational efficiency of our approach, we conduct a comprehensive comparison between TreeReasoner and naive chain-based Tool Call on the LongVideoBench under the Pass@K setting, where K denotes the ratio of token consumption relative to the naive Tool Call baseline. As shown in Table 4, TreeReasoner achieves superior performance while maintaining comparable token consumption to naive Tool Call Pass@K, demonstrating its efficiency advantage. This improvement stems from TreeReasoner's ability to effectively explore multiple "local-global" video understanding paths in parallel, thereby circumventing the accumulated errors and global misjudgments that often arise from naive Tool Call's sequential, single-path exploration strategy.

5. Conclusion

In this work, we present TreeReasoner, a novel framework that recasts long-video understanding as an active search problem over temporal hypotheses. By maintaining multiple parallel reasoning trajectories in a tree structure and strategically invoking frame-level tools—temporal zooming, jumping, and sliding—our approach efficiently discovers minimal evidence chains without exhaustive frame processing. Trained end-to-end via *ToT-RPO*, TreeReasoner achieves state-of-the-art performance across six challenging benchmarks while requiring substantially fewer frames than existing methods. Using MiMo-VL 7B as example, our experiments demonstrate consistent improvements of 7.3% over corresponding standard IO version and 2.1% over naive chain-based tool-calling approaches with training, validating that tree-structured exploration with tool augmentation provides a principled and efficient solution for long-video understanding. The interpretable emergent behaviors—including temporal bracketing and adaptive resolution adjustment—further confirm that our framework learns sophisticated temporal reasoning strategies that mirror human video comprehension processes.

References

[1] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025. 1, 6, 7

[2] Boyu Chen, Zhengrong Yue, Siran Chen, Zikang Wang, Yang Liu, Peng Li, and Yali Wang. Lvagent: Long video understanding by multi-round dynamical collaboration of mllm agents. *arXiv preprint arXiv:2503.10200*, 2025. 1, 2

[3] Xinlong Chen, Yuanxing Zhang, Yushuo Guan, Weihong Lin, Zekun Wang, Bohan Zeng, Yang Shi, Sihan Yang, Qiang Liu, Pengfei Wan, Liang Wang, and Tieniu Tan. Vidbridge-r1: Bridging qa and captioning for rl-based video understanding models with intermediate proxy tasks, 2025. 2

[4] Yukang Chen, Wei Huang, Baifeng Shi, Qinghao Hu, Hanrong Ye, Ligeng Zhu, Zhijian Liu, Pavlo Molchanov, Jan Kautz, Xiaojuan Qi, Sifei Liu, Hongxu Yin, Yao Lu, and Song Han. Scaling rl to long videos. 2025. 14

[5] Yukang Chen, Wei Huang, Baifeng Shi, Qinghao Hu, Hanrong Ye, Ligeng Zhu, Zhijian Liu, Pavlo Molchanov, Jan Kautz, Xiaojuan Qi, et al. Scaling rl to long videos. *arXiv preprint arXiv:2507.07966*, 2025. 2

[6] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025. 6

[7] Jisheng Dang, Jingze Wu, Teng Wang, Xuanhui Lin, Nannan Zhu, Hongbo Chen, Wei-Shi Zheng, Meng Wang, and Tat-Seng Chua. Reinforcing video reasoning with focused thinking. *arXiv preprint arXiv:2505.24718*, 2025. 2

[8] DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. 2, 5

[9] Jiajun Fei, Dian Li, Zhidong Deng, Zekun Wang, Gang Liu, and Hui Wang. Video-ccam: Enhancing video-language understanding with causal cross-attention masks for short and long videos. *arXiv preprint arXiv:2408.14023*, 2024. 1, 2

[10] Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Junfei Wu, Xiaoying Zhang, Benyou Wang, and Xiangyu Yue. Video-r1: Reinforcing video reasoning in mllms. *arXiv preprint arXiv:2503.21776*, 2025. 2

[11] Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Caifeng Shan, Ran He, and Xing Sun. Video-mme: The first-ever comprehensive evaluation benchmark of multimodal llms in video analysis, 2025. 7

[12] Hongcheng Gao, Jiashu Qu, Jingyi Tang, Baolong Bi, Yue Liu, Hongyu Chen, Li Liang, Li Su, and Qingming Huang. Exploring hallucination of large multimodal models in video understanding: Benchmark, analysis and mitigation, 2025. 2

[13] Google. Gemini 2.5 pro: A multimodal reasoning model for video, audio, and text. `.https://deepmind.google/models/gemini/pro/`, 2025. 1, 2, 14

[14] Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang, Jianyu Jiang, Jiawei Wang, Jingji Chen, Jingjia Huang, Kang Lei, Liping Yuan, Lishu Luo, Pengfei Liu, Qinghao Ye, Rui Qian, Shen Yan, Shixiong Zhao, Shuai Peng, Shuangye Li, Sihang Yuan, Sijin Wu, Tianheng Cheng, Weiwei Liu, Wenqian Wang, Xianhan Zeng, Xiao Liu, Xiaobo Qin, Xiaohan Ding, Xiaojun Xiao, Xiaoying Zhang, Xuanwei Zhang, Xuehan Xiong, Yanghua Peng, Yangrui Chen, Yanwei Li, Yanxu Hu, Yi Lin, Yiyuan Hu, Yiyuan Zhang, Youbin Wu, Yu Li, Yudong Liu, Yue Ling, Yujia Qin, Zanbo Wang, Zhiwu He, Aoxue Zhang,

Bairen Yi, Bencheng Liao, Can Huang, Can Zhang, Chaorui Deng, Chaoyi Deng, Cheng Lin, Cheng Yuan, Chenggang Li, Chenhui Gou, Chenwei Lou, Chengzhi Wei, Chundian Liu, Chunyuan Li, Deyao Zhu, Donghong Zhong, Feng Li, Feng Zhang, Gang Wu, Guodong Li, Guohong Xiao, Haibin Lin, Haihua Yang, Haoming Wang, Heng Ji, Hongxiang Hao, Hui Shen, Huixia Li, Jiahao Li, Jialong Wu, Jianhua Zhu, Jianpeng Jiao, Jiashi Feng, Jiaze Chen, Jianhui Duan, Jihao Liu, Jin Zeng, Jingqun Tang, Jingyu Sun, Joya Chen, Jun Long, Junda Feng, Junfeng Zhan, Junjie Fang, Junting Lu, Kai Hua, Kai Liu, Kai Shen, Kaiyuan Zhang, Ke Shen, Ke Wang, Keyu Pan, Kun Zhang, Kunchang Li, Lanxin Li, Lei Li, Lei Shi, Li Han, Liang Xiang, Liangqiang Chen, Lin Chen, Lin Li, Lin Yan, Liying Chi, Longxiang Liu, Mengfei Du, Mingxuan Wang, Ningxin Pan, Peibin Chen, Pengfei Chen, Pengfei Wu, Qingqing Yuan, Qingyao Shuai, Qiuyan Tao, Renjie Zheng, Renrui Zhang, Ru Zhang, Rui Wang, Rui Yang, Rui Zhao, Shaoqiang Xu, Shihao Liang, Shipeng Yan, Shu Zhong, Shuaishuai Cao, Shuangzhi Wu, Shufan Liu, Shuhan Chang, Songhua Cai, Tenglong Ao, Tianhao Yang, Tingting Zhang, Wanjun Zhong, Wei Jia, Wei Weng, Weihao Yu, Wenhao Huang, Wenjia Zhu, Wenli Yang, Wenzhi Wang, Xiang Long, XiangRui Yin, Xiao Li, Xiaolei Zhu, Xiaoying Jia, Xijin Zhang, Xin Liu, Xinchen Zhang, Xinyu Yang, Xiongcai Luo, Xiuli Chen, Xuantong Zhong, Xuefeng Xiao, Xujing Li, Yan Wu, Yawei Wen, Yifan Du, Yihao Zhang, Yining Ye, Yonghui Wu, Yu Liu, Yu Yue, Yufeng Zhou, Yufeng Yuan, Yuhang Xu, Yuhong Yang, Yun Zhang, Yunhao Fang, Yuntao Li, Yurui Ren, Yuwen Xiong, Zehua Hong, Zehua Wang, Zewei Sun, Zeyu Wang, Zhao Cai, Zhaoyue Zha, Zhecheng An, Zhehui Zhao, Zhengzhuo Xu, Zhipeng Chen, Zhiyong Wu, Zhuofan Zheng, Zihao Wang, Zilong Huang, Ziyu Zhu, and Zuquan Song. Seed1.5-vl technical report, 2025. 1

[15] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 6

[16] Chaoya Jiang, Yongrui Heng, Wei Ye, Han Yang, Haiyang Xu, Ming Yan, Ji Zhang, Fei Huang, and Shikun Zhang. Vlm-r^3: Region recognition, reasoning, and refinement for enhanced multimodal chain-of-thought, 2025. 2

[17] Xin Lai, Junyi Li, Wei Li, Tao Liu, Tianjian Li, and Hengshuang Zhao. Mini-o3: Scaling up reasoning patterns and interaction turns for visual search, 2025. 2

[18] Bangzheng Li, Ximeng Sun, Jiang Liu, Ze Wang, Jialian Wu, Xiaodong Yu, Hao Chen, Emad Barsoum, Muhao Chen, and Zicheng Liu. Latent visual reasoning, 2025. 2

[19] Xinhao Li, Yi Wang, Jiashuo Yu, Xiangyu Zeng, Yuhan Zhu, Haian Huang, Jianfei Gao, Kunchang Li, Yinan He, Chenting Wang, et al. Videochat-flash: Hierarchical compression for long-context video modeling. *arXiv preprint arXiv:2501.00574*, 2024. 1, 2

[20] Xinhao Li, Ziang Yan, Desen Meng, Lu Dong, Xiangyu Zeng, Yinan He, Yali Wang, Yu Qiao, Yi Wang, and Limin Wang. Videochat-r1: Enhancing spatio-temporal perception via reinforcement fine-tuning. *arXiv preprint arXiv:2504.06958*, 2025. 2

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

[22] Ziyu Liu, Yuhang Zang, Yushan Zou, Zijian Liang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi Wang. Visual agentic reinforcement fine-tuning, 2025. 2

[23] Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. *Advances in Neural Information Processing Systems*, 36:46212–46244, 2023. 7

[24] Damiano Marsili, Rohun Agrawal, Yisong Yue, and Georgia Gkioxari. Visual agentic ai for spatial reasoning with a dynamic api, 2025. 2

[25] Meta AI. The llama 3.2 herd: Multilingual & multimodal llms for edge and vision. https://arxiv.org/abs/2407.21783, 2024. 6

[26] OpenAI. Openai o3 and o4-mini. https://openai.com/index/introducing-o3-and-o4-mini/, 2025. 1

[27] Yingzhe Peng, Gongrui Zhang, Miaosen Zhang, Zhiyuan You, Jie Liu, Qipeng Zhu, Kai Yang, Xingzhong Xu, Xin Geng, and Xu Yang. Lmm-r1: Empowering 3b lmms with strong reasoning abilities through two-stage rule-based rl, 2025. 2

[28] Viorica Pătrăucean, Lucas Smaira, Ankush Gupta, Adrià Recasens Continente, Larisa Markeeva, Dylan Banarse, Skanda Koppula, Joseph Heyward, Mateusz Malinowski, Yi Yang, Carl Doersch, Tatiana Matejovicova, Yury Sulsky, Antoine Miech, Alex Frechette, Hanna Klimczak, Raphael Koster, Junlin Zhang, Stephanie Winkler, Yusuf Aytar, Simon Osindero, Dima Damen, Andrew Zisserman, and João Carreira. Perception test: A diagnostic benchmark for multimodal video models. In *Advances in Neural Information Processing Systems*, 2023. 14

[29] Ziyao Shangguan, Chuhan Li, Yuxuan Ding, Yanan Zheng, Yilun Zhao, Tesca Fitzgerald, and Arman Cohan. Tomato: Assessing visual temporal reasoning capabilities in multimodal foundation models, 2025. 7

[30] Wei Shen, Jiangbo Pei, Yi Peng, Xuchen Song, Yang Liu, Jian Peng, Haofeng Sun, Yunzhuo Hao, Peiyu Wang, Jianhao Zhang, and Yahui Zhou. Skywork-r1v3 technical report, 2025. 1

[31] Yan Shu, Peitian Zhang, Zheng Liu, Minghao Qin, Junjie Zhou, Tiejun Huang, and Bo Zhao. Video-xl: Extra-long vision language model for hour-scale video understanding. *arXiv preprint arXiv:2409.14485*, 2024. 1, 2

[32] Huajie Tan, Yuheng Ji, Xiaoshuai Hao, Xiansheng Chen, Pengwei Wang, Zhongyuan Wang, and Shanghang Zhang. Reason-rft: Reinforcement fine-tuning for visual reasoning of vision language models, 2025. 2

[33] Core Team, Zihao Yue, Zhenru Lin, Yifan Song, Weikun Wang, Shuhuai Ren, Shuhao Gu, Shicheng Li, Peidian Li, Liang Zhao, Lei Li, Kainan Bao, Hao Tian, Hailin Zhang, Gang Wang, Dawei Zhu, Cici, Chenhong He, Bowen Ye, Bowen Shen, Zihan Zhang, Zihan Jiang, Zhixian Zheng, Zhichao Song, Zhenbo Luo, Yue Yu, Yudong Wang, Yuanyuan Tian, Yu Tu, Yihan Yan, Yi Huang, Xu

Wang, Xinzhe Xu, Xingchen Song, Xing Zhang, Xing Yong, Xin Zhang, Xiangwei Deng, Wenyu Yang, Wenhan Ma, Weiwei Lv, Weiji Zhuang, Wei Liu, Sirui Deng, Shuo Liu, Shimao Chen, Shihua Yu, Shaohui Liu, Shande Wang, Rui Ma, Qiantong Wang, Peng Wang, Nuo Chen, Menghang Zhu, Kangyang Zhou, Kang Zhou, Kai Fang, Jun Shi, Jinhao Dong, Jiebao Xiao, Jiaming Xu, Huaqiu Liu, Hongshen Xu, Heng Qu, Haochen Zhao, Hanglong Lv, Guoan Wang, Duo Zhang, Dong Zhang, Di Zhang, Chong Ma, Chang Liu, Can Cai, and Bingquan Xia. Mimo-vl technical report, 2025. 6, 7

[34] Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025. 6

[35] Kimi Team, Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, Zhuofu Chen, Jialei Cui, Hao Ding, Mengnan Dong, Angang Du, Chenzhuang Du, Dikang Du, Yulun Du, Yu Fan, Yichen Feng, Kelin Fu, Bofei Gao, Hongcheng Gao, Peizhong Gao, Tong Gao, Xinran Gu, Longyu Guan, Haiqing Guo, Jianhang Guo, Hao Hu, Xiaoru Hao, Tianhong He, Weiran He, Wenyang He, Chao Hong, Yangyang Hu, Zhenxing Hu, Weixiao Huang, Zhiqi Huang, Zihao Huang, Tao Jiang, Zhejun Jiang, Xinyi Jin, Yongsheng Kang, Guokun Lai, Cheng Li, Fang Li, Haoyang Li, Ming Li, Wentao Li, Yanhao Li, Yiwei Li, Zhaowei Li, Zheming Li, Hongzhan Lin, Xiaohan Lin, Zongyu Lin, Chengyin Liu, Chenyu Liu, Hongzhang Liu, Jingyuan Liu, Junqi Liu, Liang Liu, Shaowei Liu, T. Y. Liu, Tianwei Liu, Weizhou Liu, Yangyang Liu, Yibo Liu, Yiping Liu, Yue Liu, Zhengying Liu, Enzhe Lu, Lijun Lu, Shengling Ma, Xinyu Ma, Yingwei Ma, Shaoguang Mao, Jie Mei, Xin Men, Yibo Miao, Siyuan Pan, Yebo Peng, Ruoyu Qin, Bowen Qu, Zeyu Shang, Lidong Shi, Shengyuan Shi, Feifan Song, Jianlin Su, Zhengyuan Su, Xinjie Sun, Flood Sung, Heyi Tang, Jiawen Tao, Qifeng Teng, Chensi Wang, Dinglu Wang, Feng Wang, Haiming Wang, Jianzhou Wang, Jiaxing Wang, Jinhong Wang, Shengjie Wang, Shuyi Wang, Yao Wang, Yejie Wang, Yiqin Wang, Yuxin Wang, Yuzhi Wang, Zhaoji Wang, Zhengtao Wang, Zhexu Wang, Chu Wei, Qianqian Wei, Wenhao Wu, Xingzhe Wu, Yuxin Wu, Chenjun Xiao, Xiaotong Xie, Weimin Xiong, Boyu Xu, Jing Xu, Jinjing Xu, L. H. Xu, Lin Xu, Suting Xu, Weixin Xu, Xinran Xu, Yangchuan Xu, Ziyao Xu, Junjie Yan, Yuzi Yan, Xiaofei Yang, Ying Yang, Zhen Yang, Zhilin Yang, Zonghan Yang, Haotian Yao, Xingcheng Yao, Wenjie Ye, Zhuorui Ye, Bohong Yin, Longhui Yu, Enming Yuan, Hongbang Yuan, Mengjie Yuan, Haobing Zhan, Dehao Zhang, Hao Zhang, Wanlu Zhang, Xiaobin Zhang, Yangkun Zhang, Yizhi Zhang, Yongting Zhang, Yu Zhang, Yutao Zhang, Yutong Zhang, Zheng Zhang, Haotian Zhao, Yikai Zhao, Huabin Zheng, Shaojie Zheng, Jianren Zhou, Xinyu Zhou, Zaida Zhou, Zhen Zhu, Weiyu Zhuang, and Xinxing Zu. Kimi k2: Open agentic intelligence, 2025. 2

[36] Kimi Team, Angang Du, Bohong Yin, Bowei Xing, Bowen Qu, Bowen Wang, Cheng Chen, Chenlin Zhang, Chenzhuang Du, Chu Wei, Congcong Wang, Dehao Zhang, Dikang Du, Dongliang Wang, Enming Yuan, Enzhe Lu, Fang Li, Flood Sung, Guangda Wei, Guokun Lai, Han Zhu, Hao Ding, Hao Hu, Hao Yang, Hao Zhang, Haoning Wu, Haotian Yao, Haoyu Lu, Heng Wang, Hongcheng Gao, Huabin Zheng, Jiaming Li, Jianlin Su, Jianzhou Wang, Jiaqi Deng, Jiezhong Qiu, Jin Xie, Jinhong Wang, Jingyuan Liu, Junjie Yan, Kun Ouyang, Liang Chen, Lin Sui, Longhui Yu, Mengfan Dong, Mengnan Dong, Nuo Xu, Pengyu Cheng, Qizheng Gu, Runjie Zhou, Shaowei Liu, Sihan Cao, Tao Yu, Tianhui Song, Tongtong Bai, Wei Song, Weiran He, Weixiao Huang, Weixin Xu, Xiaokun Yuan, Xingcheng Yao, Xingzhe Wu, Xinhao Li, Xinxing Zu, Xinyu Zhou, Xinyuan Wang, Y. Charles, Yan Zhong, Yang Li, Yangyang Hu, Yanru Chen, Yejie Wang, Yibo Liu, Yibo Miao, Yidao Qin, Yimin Chen, Yiping Bao, Yiqin Wang, Yongsheng Kang, Yuanxin Liu, Yuhao Dong, Yulun Du, Yuxin Wu, Yuzhi Wang, Yuzi Yan, Zaida Zhou, Zhaowei Li, Zhejun Jiang, Zheng Zhang, Zhilin Yang, Zhiqi Huang, Zihao Huang, Zijia Zhao, Ziwei Chen, and Zongyu Lin. Kimi-vl technical report, 2025. 1

[37] V Team, Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, Shuaiqi Duan, Weihan Wang, Yan Wang, Yean Cheng, Zehai He, Zhe Su, Zhen Yang, Ziyang Pan, Aohan Zeng, Baoxu Wang, Bin Chen, Boyan Shi, Changyu Pang, Chenhui Zhang, Da Yin, Fan Yang, Guoqing Chen, Jiazheng Xu, Jiale Zhu, Jiali Chen, Jing Chen, Jinhao Chen, Jinghao Lin, Jinjiang Wang, Junjie Chen, Leqi Lei, Letian Gong, Leyi Pan, Mingdao Liu, Mingde Xu, Mingzhi Zhang, Qinkai Zheng, Sheng Yang, Shi Zhong, Shiyu Huang, Shuyuan Zhao, Siyan Xue, Shangqin Tu, Shengbiao Meng, Tianshu Zhang, Tianwei Luo, Tianxiang Hao, Tianyu Tong, Wenkai Li, Wei Jia, Xiao Liu, Xiaohan Zhang, Xin Lyu, Xinyue Fan, Xuancheng Huang, Yanling Wang, Yadong Xue, Yanfeng Wang, Yanzi Wang, Yifan An, Yifan Du, Yiming Shi, Yiheng Huang, Yilin Niu, Yuan Wang, Yuanchang Yue, Yuchen Li, Yutao Zhang, Yuting Wang, Yu Wang, Yuxuan Zhang, Zhao Xue, Zhenyu Hou, Zhengxiao Du, Zihan Wang, Peng Zhang, Debing Liu, Bin Xu, Juanzi Li, Minlie Huang, Yuxiao Dong, and Jie Tang. Glm-4.5v and glm-4.1v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning, 2025. 1

[38] Haozhe Wang, Alex Su, Weiming Ren, Fangzhen Lin, and Wenhu Chen. Pixel reasoner: Incentivizing pixel-space reasoning with curiosity-driven reinforcement learning, 2025. 2

[39] Kangrui Wang, Pingyue Zhang, Zihan Wang, Yaning Gao, Linjie Li, Qineng Wang, Hanyang Chen, Chi Wan, Yiping Lu, Zhengyuan Yang, Lijuan Wang, Ranjay Krishna, Jiajun Wu, Li Fei-Fei, Yejin Choi, and Manling Li. Vagen: Reinforcing world model reasoning for multi-turn vlm agents, 2025. 2

[40] Qi Wang, Yanrui Yu, Ye Yuan, Rui Mao, and Tianfei Zhou. Videorft: Incentivizing video reasoning capability in mllms via reinforced fine-tuning. *arXiv preprint arXiv:2505.12434*, 2025. 2

[41] Weiyun Wang, Zhangwei Gao, Lianjie Chen, Zhe Chen, Jinguo Zhu, Xiangyu Zhao, Yangzhou Liu, Yue Cao, Shenglong

Ye, Xizhou Zhu, Lewei Lu, Haodong Duan, Yu Qiao, Jifeng Dai, and Wenhai Wang. Visualprm: An effective process reward model for multimodal reasoning, 2025. 2

[42] Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025. 1

[43] Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. Videoagent: Long-form video understanding with large language model as agent. In *European Conference on Computer Vision*, pages 58–76. Springer, 2024. 1, 2

[44] Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Zun Wang, Yansong Shi, et al. Internvideo2: Scaling foundation models for multimodal video understanding. In *European Conference on Computer Vision*, pages 396–416. Springer, 2024. 1

[45] Yuxuan Wang, Yueqian Wang, Dongyan Zhao, Cihang Xie, and Zilong Zheng. Videohallucer: Evaluating intrinsic and extrinsic hallucinations in large video-language models, 2024. 8

[46] Yi Wang, Xinhao Li, Ziang Yan, Yinan He, Jiashuo Yu, Xiangyu Zeng, Chenting Wang, Changlian Ma, Haian Huang, Jianfei Gao, et al. Internvideo2. 5: Empowering video mllms with long and rich context modeling. *arXiv preprint arXiv:2501.12386*, 2025. 1, 2

[47] Zikang Wang, Boyu Chen, Zhengrong Yue, Yi Wang, Yu Qiao, Limin Wang, and Yali Wang. Videochat-a1: Thinking with long videos by chain-of-shot reasoning. *arXiv preprint arXiv:2506.06097*, 2025. 1, 2

[48] Zhenhailong Wang, Xuehang Guo, Sofia Stoica, Haiyang Xu, Hongru Wang, Hyeonjeong Ha, Xiusi Chen, Yangyi Chen, Ming Yan, Fei Huang, and Heng Ji. Perception-aware policy optimization for multimodal reasoning, 2025. 2

[49] 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. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3272–3283, 2025. 1, 2

[50] Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. Longvideobench: A benchmark for long-context interleaved video-language understanding, 2024. 7

[51] Jiaer Xia, Yuhang Zang, Peng Gao, Sharon Li, and Kaiyang Zhou. Visionary-r1: Mitigating shortcuts in visual reasoning with reinforcement learning, 2025. 2

[52] 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 (CVPR)*, pages 9777–9786, 2021. 14

[53] Guowei Xu, Peng Jin, Ziang Wu, Hao Li, Yibing Song, Lichao Sun, and Li Yuan. Llava-cot: Let vision language models reason step-by-step, 2025. 2

[54] Yi Xu, Chengzu Li, Han Zhou, Xingchen Wan, Caiqi Zhang, Anna Korhonen, and Ivan Vulić. Visual planning: Let's think only with images, 2025. 2

[55] Fuzhao Xue, Yukang Chen, Dacheng Li, Qinghao Hu, Ligeng Zhu, Xiuyu Li, Yunhao Fang, Haotian Tang, Shang Yang, Zhijian Liu, et al. Longvila: Scaling long-context visual language models for long videos. *arXiv preprint arXiv:2408.10188*, 2024. 1, 2

[56] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. 2, 6, 7

[57] Biao Yang, Bin Wen, Boyang Ding, Changyi Liu, Chenglong Chu, Chengru Song, Chongling Rao, Chuan Yi, Da Li, Dunju Zang, Fan Yang, Guorui Zhou, Guowang Zhang, Han Shen, Hao Peng, Haojie Ding, Hao Wang, Haonan Fan, Hengrui Ju, Jiaming Huang, Jiangxia Cao, Jiankang Chen, Jingyun Hua, Kaibing Chen, Kaiyu Jiang, Kaiyu Tang, Kun Gai, Muhao Wei, Qiang Wang, Ruitao Wang, Sen Na, Shengnan Zhang, Siyang Mao, Sui Huang, Tianke Zhang, Tingting Gao, Wei Chen, Wei Yuan, Xiangyu Wu, Xiao Hu, Xingyu Lu, Yi-Fan Zhang, Yiping Yang, Yulong Chen, Zeyi Lu, Zhenhua Wu, Zhixin Ling, Zhuoran Yang, Ziming Li, Di Xu, Haixuan Gao, Hang Li, Jing Wang, Lejian Ren, Qigen Hu, Qianqian Wang, Shiyao Wang, Xinchen Luo, Yan Li, Yuhang Hu, and Zixing Zhang. Kwai keye-vl 1.5 technical report, 2025. 1, 2

[58] Jihan Yang, Shusheng Yang, Anjali W. Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces, 2025. 7

[59] Zeyuan Yang, Xueyang Yu, Delin Chen, Maohao Shen, and Chuang Gan. Machine mental imagery: Empower multimodal reasoning with latent visual tokens, 2025. 2

[60] Linli Yao, Haoning Wu, Kun Ouyang, Yuanxing Zhang, Caiming Xiong, Bei Chen, Xu Sun, and Junnan Li. Generative frame sampler for long video understanding. *arXiv preprint arXiv:2503.09146*, 2025. 1, 2

[61] Jinhui Ye, Zihan Wang, Haosen Sun, Keshigeyan Chandrasegaran, Zane Durante, Cristobal Eyzaguirre, Yonatan Bisk, Juan Carlos Niebles, Ehsan Adeli, Li Fei-Fei, et al. Rethinking temporal search for long-form video understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 8579–8591, 2025. 1, 2

[62] Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B. Tenenbaum. Clevrer: Collision events for video representation and reasoning, 2020. 14

[63] En Yu, Kangheng Lin, Liang Zhao, Jisheng Yin, Yana Wei, Yuang Peng, Haoran Wei, Jianjian Sun, Chunrui Han, Zheng

Ge, Xiangyu Zhang, Daxin Jiang, Jingyu Wang, and Wenbing Tao. Perception-r1: Pioneering perception policy with reinforcement learning, 2025. 2

[64] 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. *CoRR*, abs/2406.16852, 2024. 1, 2

[65] Xintong Zhang, Zhi Gao, Bofei Zhang, Pengxiang Li, Xiaowen Zhang, Yang Liu, Tao Yuan, Yuwei Wu, Yunde Jia, Song-Chun Zhu, and Qing Li. Chain-of-focus: Adaptive visual search and zooming for multimodal reasoning via rl, 2025. 2

[66] Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Video instruction tuning with synthetic data, 2024. 14

[67] Ziwei Zheng, Michael Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, Chao Shen, and Xing Yu. Deepeyes: Incentivizing "thinking with images" via reinforcement learning, 2025. 2

[68] Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Zhengyang Liang, Shitao Xiao, Minghao Qin, Xi Yang, Yongping Xiong, Bo Zhang, Tiejun Huang, and Zheng Liu. Mlvu: Benchmarking multi-task long video understanding, 2025. 7

[69] Orr Zohar, Xiaohan Wang, Yonatan Bitton, Idan Szpektor, and Serena Yeung-levy. Video-star: Self-training enables video instruction tuning with any supervision. In *arXiv preprint arXiv:2407.06189*, 2024. 14

TreeReasoner: Reinforcing Tool-Augmented Tree-of-Videos Reasoning

Supplementary Material

A. Case Study

In the introduction, we posited that our agentic framework enhances reasoning by actively navigating the video timeline—a departure from passive frame processing—and highlighted the interpretability of its search strategies. In this section, we provide qualitative examples (Fig. 7 and 8) to substantiate these claims, specifically demonstrating the model's search behaviors, such as temporal bracketing, causal chain verification, and adaptive temporal resolution adjustment.

Temporal bracketing. As shown in Fig. 7a, consider a query requiring multi-step reasoning, such as "Identify the color of the pants worn by the young woman talking to an elderly man on the subway." Initially, TreeReasoner employs a coarse-grained search with a large temporal stride to rapidly scan the entire timeline, identifying broad temporal regions where key objects (e.g., subway scenes, elderly man) appear. Upon detecting these relevant contexts, the model automatically switches to a fine-grained temporal mode, progressively narrowing down the time window by examining adjacent segments. This allows it to precisely bracket the specific temporal span where the target event—"a young woman conversing with an elderly man"—occurs. Within this bracketed time window, the model then analyzes the fine-grained visual details to extract the answer. By "bracketing" the event temporally in this coarse-to-fine manner, the model avoids processing the redundant background footage, directly contributing to the efficiency gains mentioned in our experiments. Another similar case can be found in Fig. 8a.

Causal chain verification. For complex queries such as "Find the color of the pants worn by the young girl who talks to the old man on the subway", the model exhibits a hierarchical reasoning strategy. TreeReasoner first searches for video segments containing the objects mentioned in the query, including segments with subway scenes, old men, and young girls. After identifying segments with the specified objects, TreeReasoner further examines the consistency between video content and the query-specified events, ultimately pinpointing the segment where "the young girl talks to the old man" occurs. It then comprehends the object information in this segment and provides the answer. This search strategy enables TreeReasoner to retrieve query-relevant video segments at a relatively fine-grained level, forming a complete chain of visual evidence and delivering accurate responses grounded in fine-grained visual details. This behavior demonstrates that TreeReasoner validates relationships by progressively refining its search through multiple verification steps, ensuring the answer is grounded in visual evidence rather than hallucination. Another similar case can be found in Fig. 8b.

Adaptive temporal resolution adjustment. In the final reasoning tree, TreeReasoner exhibits variable temporal resolutions across different video segments through adaptive tool invocation. As shown in Fig. 7c, for static, repetitive, or query-irrelevant portions of the video, TreeReasoner dynamically increases its temporal stride, skipping large chunks of uninformative frames. Conversely, for segments closely related to the query—such as those containing the target individuals and their interaction—the model immediately invokes tools to extract short video clips and applies much smaller step sizes to capture high-frequency details of the girl's appearance and clothing. This adaptive mechanism explains how our method achieves superior accuracy while processing substantially fewer frames than fixed-rate baselines. Another similar case can be found in Fig. 8c.

B. Dataset Details

To endow TreeReasoner with robust tool-use capabilities and ensure generalization across diverse visual domains, we constructed a unified training corpus by aggregating queries from multiple open-source datasets, including CLEVRER [62], LLaVA-Video-178K [66], NExT-QA [52], PerceptionTest [28], Video-STaR [69], and LongVideo-Reason [4].

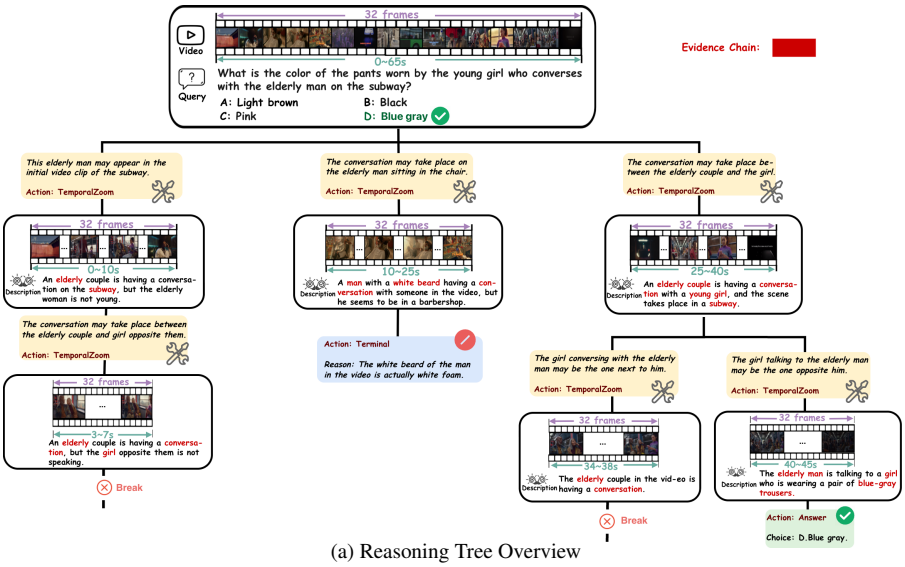
SFT Data Construction. From this broad collection, we first sampled a balanced subset of 20,000 video-question pairs to serve as the source for Supervised Fine-Tuning (SFT). Since these datasets lack explicit annotations for tool-augmented reasoning chains, we employed a knowledge distillation strategy utilizing Gemini-2.5-Pro [13] as the teacher model. Specifically, we prompted the teacher to generate multi-branch Tree-of-Tool reasoning paths based on our defined toolset (zoom, jump, slide). To align with our framework's objective of identifying a *minimal yet sufficient* chain of evidence, we applied an *efficiency-oriented filtering strategy*. For each query, we ranked the successful reasoning paths by length and retained up to the top-k ($k = 2$) shortest trajectories. Additionally, we explicitly preserved a subset of trajectories where the model actively triggered termination to ensure robust exploration capabilities. This process resulted in a final corpus of 33,724 high-quality trajectories for behavioral cloning.

RL Data Construction. Distinct from the SFT dataset, we

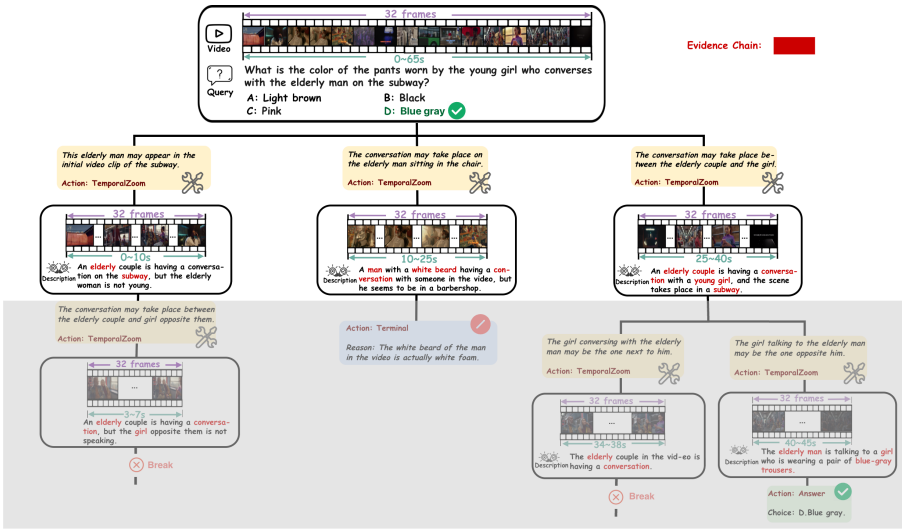
sampled an *additional, non-overlapping subset* of 40,000 video-question pairs from the aggregated corpus to drive the RL stage. By using raw queries instead of teacher trajectories, the model generates its own reasoning paths optimized via ToT-RPO, balancing accuracy with efficiency and enhancing generalization.

C. Limitations

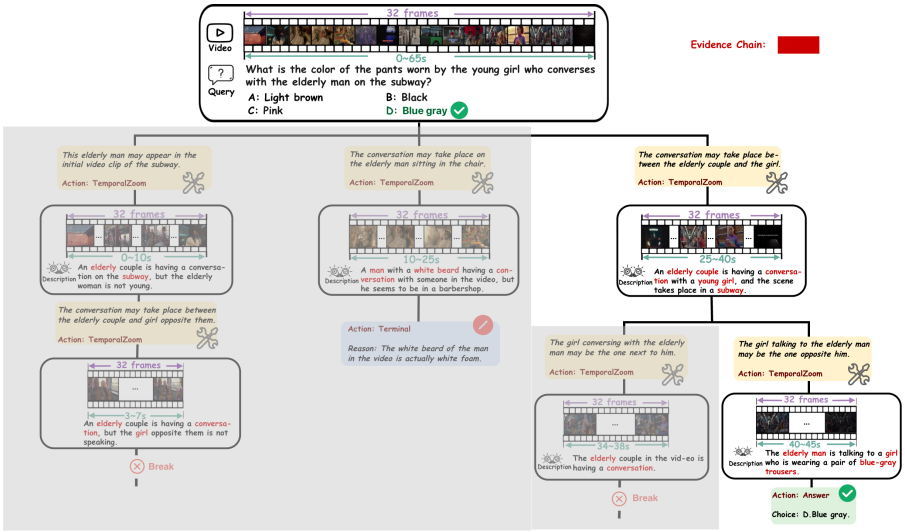
Despite the demonstrated effectiveness of TreeReasoner in handling complex video reasoning tasks, we acknowledge several limitations. First, the reliance on a tree search mechanism introduces a computational overhead compared to standard end-to-end IO of MLLMs. While our Reinforcement Learning strategy explicitly optimizes for shorter reasoning paths, the inference process still requires multiple forward passes to explore and prune branches, which may hinder deployment in strictly real-time scenarios. Second, our current toolset is limited to visual navigation primitives (zoom, jump, slide). While these are sufficient for spatiotemporal grounding, the model lacks tools for processing other modalities (e.g., audio analysis). Finally, our training paradigm depends on knowledge distillation from a teacher model (Gemini-2.5-Pro). Although we apply rigorous filtering to the distilled data, the student model's upper bound is inevitably influenced by the teacher's capabilities.



(a) Reasoning Tree Overview

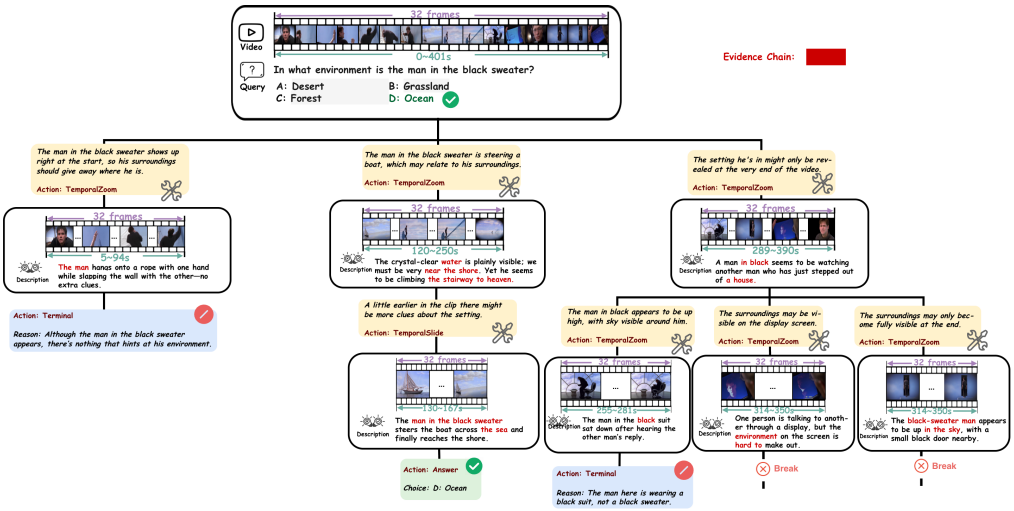


(b) Temporal Bracketing

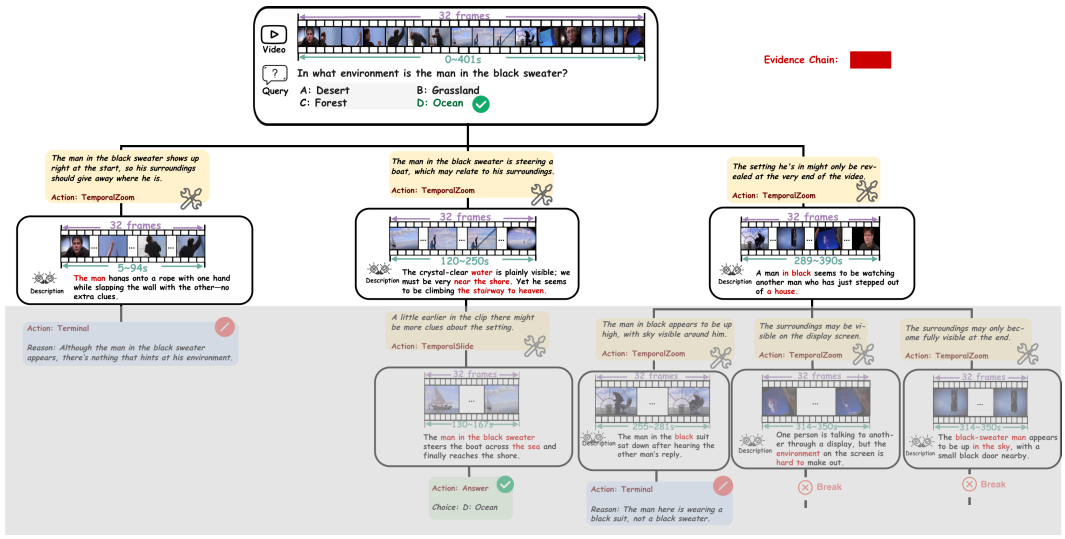


(c) Causal Chain Verification

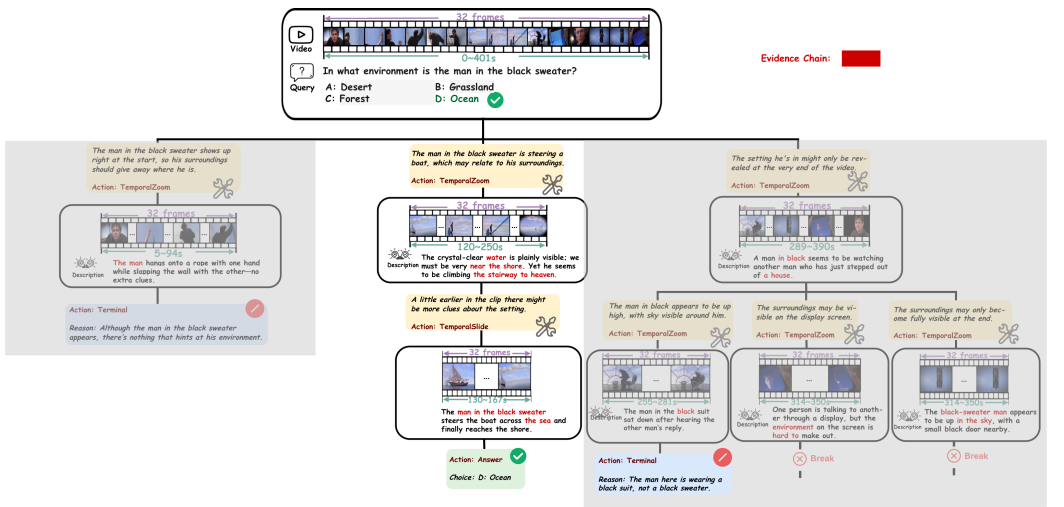
Figure 7. A case study on "identifying a girl's pants color query in subway scene": (a) shows an overview of the complete reasoning tree; (b) illustrates the temporal bracketing mechanism; (c) presents a branch trajectory along with its corresponding visual evidence chain. Irrelevant regions are masked in (b) and (c) to highlight key content.



(a) Reasoning Tree Overview



(b) Temporal Bracketing



(c) Causal Chain Verification

Figure 8. A case study on "the environment query of a man in black sweater": (a) shows an overview of the complete reasoning tree; (b) illustrates the temporal bracketing mechanism; (c) presents a branch trajectory along with its corresponding visual evidence chain. Irrelevant regions are masked in (b) and (c) to highlight key content.