VIDEOSEEKER: INCENTIVIZING INSTANCE-LEVEL VIDEO UNDERSTANDING VIA NATIVE AGENTIC TOOL INVOCATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large Vision-Language Models (LVLMs) have shown significant progress in video understanding, yet they face substantial challenges in tasks requiring precise spatiotemporal localization at the instance level. Existing methods primarily rely on text prompts for human-model interaction, but these prompts struggle to provide precise spatial and temporal references, resulting in poor user experience. Furthermore, current approaches typically decouple visual perception from language reasoning, centering reasoning around language rather than visual content, which limits the model's ability to proactively perceive fine-grained visual evidence. To address these challenges, we propose **VideoSeeker**, a novel paradigm for instance-level video understanding through visual prompts. VideoSeeker seamlessly integrates agentic reasoning with instance-level video understanding tasks, enabling the model to proactively perceive and retrieve relevant video segments on demand. We construct a four-stage fully automated data synthesis pipeline to efficiently generate large-scale, high-quality instance-level video data. We internalize tool-calling and proactive perception capabilities into the model via cold-start supervision and RL training, building a powerful video understanding model. Experiments demonstrate that our model achieves an average improvement of +13.7% over baselines on instance-level video understanding tasks, surpassing powerful closed-source models such as GPT-4o and Gemini-2.5-Pro, while also showing effective transferability on general video understanding benchmarks. The relevant datasets and code will be released publicly.

1 INTRODUCTION

Large Vision Language Models (LVLMs) have achieved significant progress in recent years, demonstrating exceptional capabilities across diverse tasks including image captioning (Zeng et al., 2025b; Deitke et al., 2025; Xing et al., 2025; Clark et al., 2026), visual question answering (Chen et al., 2024a; Bai et al., 2025; Zeng et al., 2025a; Xu et al., 2025; Chen et al., 2024b), video understanding (Zhao et al., 2025b; Fu et al., 2025; Qi et al., 2025; Hong et al., 2026; Wang et al., 2025e; Ren et al., 2024), and complex multimodal reasoning (Team et al., 2026; Chen et al., 2025a). Recently, methods (Feng et al., 2025; Wang et al., 2025c;d) have introduced reinforcement learning (RL) into video question answering and temporal localization, leveraging environmental reward signals to guide models in exploring superior reasoning strategies and achieving remarkable improvements in video understanding.

However, existing methods still suffer from two key limitations. *(1) Most current approaches decouple visual perception from language reasoning*, centering reasoning on language rather than visual evidence (Feng et al., 2025; Wang et al., 2025c;d), which weakens visual grounding and often causes hallucinations in long-video scenarios (Yang et al., 2025b). Moreover, the widely used single-pass uniform sampling is a passive perception mechanism that cannot adaptively capture key visual evidence, frequently missing fine-grained details critical for reasoning (Fu et al., 2025); such methods therefore struggle with precise localization, e.g., identifying when a person appears for the second time. *(2) Existing methods and benchmarks mainly focus on holistic video understanding* (Fu et al., 2025; Wu et al., 2024), emphasizing global semantics and coarse-grained events while lacking fine-grained spatio-temporal localization and reasoning for specific instances (Wang et al.,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

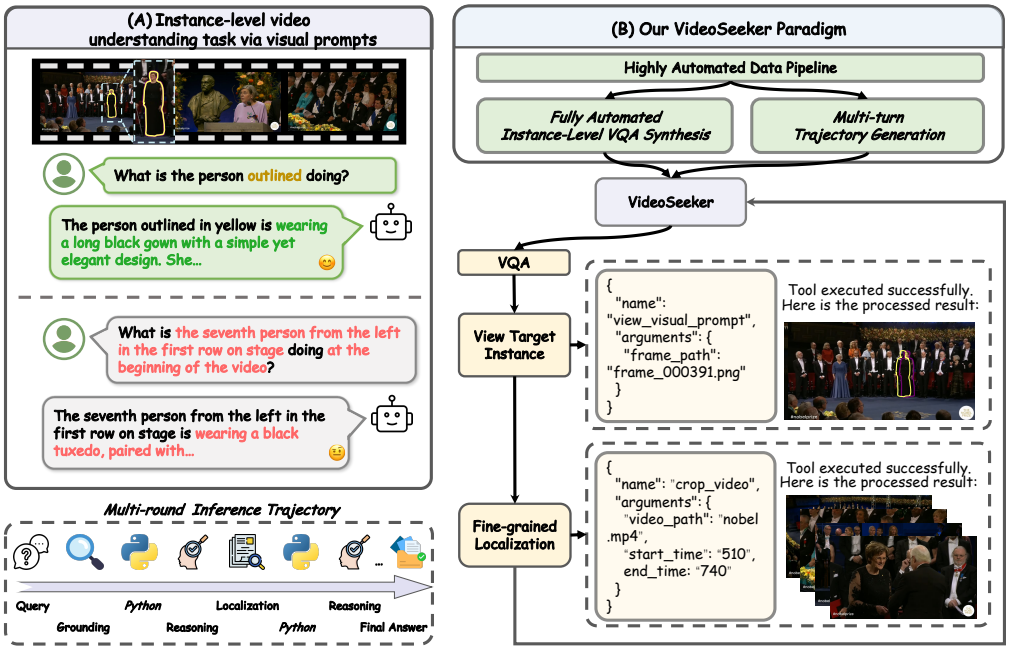

Figure 1: **Overview of VideoSeeker.** *(A): Instance-level video understanding tasks* require the model to accurately locate and reason about specific video instances via visual prompts, given a video, a visual prompt frame, and a query; vs. text-only descriptions, visual prompts enable more precise and intuitive interaction. *(B):* A four-stage pipeline constructs instance-level video data, then a two-stage training strategy integrates agentic reasoning with instance-level video understanding.

2025f). Relying solely on text queries (Figure 1. A) also precludes precise spatial-temporal references (Zhao et al., 2025b), forcing users to describe targets with lengthy referential language, degrading interaction efficiency, and hindering evaluation in complex multi-object scenarios.

To address these issues, we propose **VideoSeeker**, a novel paradigm for instance-level video understanding based on visual prompts (Figure 1. B). Unlike text descriptions, visual prompts let users directly annotate target regions on video frames, yielding more precise spatial and temporal references. To support this paradigm, we construct a four-stage fully automated data synthesis pipeline (Figure 2): a lightweight language model pre-screens QA candidates, a video understanding model verifies target uniqueness to ensure solvability, SAM3 Carion et al. (2025) produces pixel-level masks, and diverse visual prompts are finally rendered into training-ready instance-level video QA data. Building on this data, we adopt a two-stage strategy of SFT cold-start followed by Agentic RL, which drives the model to explore the policy space with high information gain and integrates multi-round agentic reasoning and instance-level video understanding into the base model. Extensive experiments show that VideoSeeker significantly outperforms all open-source baselines on the instance-level benchmark V2P-Bench: our 8B model achieves an average improvement of +13.7% over the baseline, surpassing closed-source models such as GPT-4o and Gemini-2.5-Pro, while transferring effectively to general video understanding scenarios.

In a nutshell, our contributions are as follows:

- We propose VideoSeeker, an agentic instance-level video understanding paradigm. By organically integrating agentic reasoning, VideoSeeker breaks through the limitations of text queries and achieves more precise references.
- We construct a four-stage instance-level video question answering data synthesis pipeline and efficiently generates large-scale, high-quality instance-level video data, providing an effective solution to the scarcity of relevant training data.
- Extensive experiments demonstrate that VideoSeeker significantly outperforms all open-source and proprietary baselines on instance-level video understanding tasks, while also exhibiting effective transferability to general video understanding scenarios.

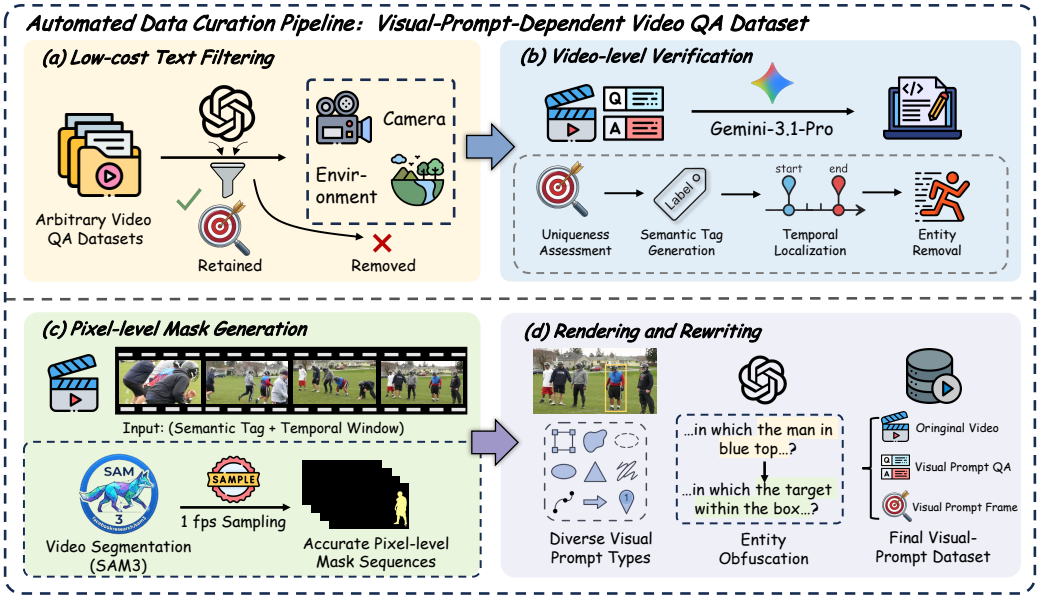

Figure 2: **Our Data Pipeline.** *(1) Low-cost Text Filtering* rapidly filters pure text QA pairs; *(2) Video-level Verification* verifies target uniqueness and generates semantic tags; *(3) Pixel-level Mask Generation* produces pixel-wise masks via SAM3; *(4) Visual Prompt Rendering* renders diverse visual prompt types and rewrites QA to depend on them.

2 Related Works

Reinforcement Learning for Vision Language Models. Inspired by the success of large reasoning models such as OpenAI o1 (Jaech et al., 2024) and DeepSeek-R1 (Guo et al., 2025), recent studies extend GRPO-style RL (Shao et al., 2024) from text-only reasoning to multimodal domains (Rafailov et al., 2023). In vision, methods enhance reasoning for image QA (Huang et al., 2025; Meng et al., 2025; Deng et al., 2025), grounding (Liu et al., 2025; Shen et al., 2025). For example, Perception-R1 (Yu et al., 2025) leverages object matching and IoU as reward signals to improve grounding, and DeepEyes (Zheng et al., 2025) shows how RL can encourage models to invoke visual tools, thereby expanding perceptual abilities. Video-centric approaches further tackle temporal reasoning tasks such as video QA (Feng et al., 2025; Wang et al., 2025c) and temporal grounding (Wang et al., 2025f; Li et al., 2025), with Video-R1 (Feng et al., 2025), VideoChat-R1 (Li et al., 2025) and VideoRFT (Wang et al., 2025c) being representative works. Additionally, Vision-R1 (Huang et al., 2025) and R1-OneVision (Yang et al., 2025a) construct multimodal CoT datasets by converting visual information into textual representations to support stronger reasoning. Despite these advances, most methods still rely on text-based CoT reasoning (Feng et al., 2025; Li et al., 2025; Chen et al., 2025b), which remains largely language-centric (Yang et al., 2025b), limiting visual reasoning and increasing hallucinations in long-video scenarios. This motivates us to explore how to enable more effective video reasoning through visual tool augmentation.

Tool-Augmented Agentic Vision Language Models. Recent advances in LVLMs show that equipping models with external tools can enhance capabilities beyond pure text understanding and generation (Wang et al., 2025b; Zheng et al., 2025). In the image domain, methods (Zheng et al., 2025; Wang et al., 2025b; Team, 2025; Wang et al., 2025a; Hong et al., 2025) enable MLLMs to "think with images" by integrating visual tools for image reasoning, while VILA-SR (Wu et al., 2025) reinforces spatial reasoning with interwoven visual drawing. In the video domain, LongVT (Yang et al., 2025b) proposes iMCoTT that enables MLLMs to perform native temporal retrieval and reasoning by dynamically selecting and re-inspecting relevant video segments, without an auxiliary retriever. VITAL (Zhang et al., 2025) constructs a visual toolbox that allows models to densely sample new video frames on demand during reasoning, enabling precise long video reasoning. Additionally, Ego-R1 (Tian et al., 2025) explores chain-of-tool-thought reasoning in first-person videos, and PyVision (Zhao et al., 2025a) proposes dynamic tool calling. However, our method differs from prior works such as LongVT (Yang et al., 2025b) and VITAL (Zhang et al., 2025) in the

following key aspects: *(1) VideoSeeker targets instance-level video understanding tasks*, focusing on precise localization and tracking of specific target instances within videos; whereas LongVT and VITAL primarily emphasize holistic semantic modeling. *(2) VideoSeeker employs visual prompts (e.g., bounding boxes, points, and masks) as queries*, enabling direct specification of target instances with more precise spatial and temporal references; whereas prior works rely entirely on pure text queries, requiring extensive referential language to describe targets. *(3) We design a four-stage fully automated data pipeline* that efficiently generates large-scale, high-quality instance-level video data, and propose a two-stage training paradigm to internalize native tool-calling capabilities into the base model, enabling native instance-level video understanding.

3 METHOD

3.1 TASK FORMULATION AND ENVIRONMENTAL INTERACTION

Task Formulation. Given a query Q, a visual prompt frame $\mathcal{F}_{vp}$ and a video $\mathcal{V}$ of arbitrary length, the goal of instance-level video understanding is to accurately answer the query Q with respect to the specific instance indicated by $\mathcal{F}_{vp}$, and output a grounded answer A. Unlike general video question answering where the answer is independent of a particular object, instance-level video understanding requires the model to *(1)* precisely associate the visual prompt with the corresponding target instance in $\mathcal{V}$ and *(2)* reason about the temporal dynamics of that specific instance across $\mathcal{V}$ to produce the final answer A.

Environmental Interaction. The policy model π_θ interacts with the video environment through multi-turn active perception control, rather than passively encoding all context in a single pass. Specifically, the model is equipped with a perception tool set $\mathcal{T} = \{view_visual_prompt,\ crop_video\}$: the former continuously provides visual prompt frames $\mathcal{F}_{vp}$, maintaining a cognitive anchor of the target instance appearance throughout reasoning; the latter endows the model with fine-grained local observation capability, enabling active filtering of keyframes and removal of redundant information when processing long videos with complex visual prompts. The two tools are formally defined as:

$$\mathcal{I}_{vp} = \texttt{view_visual_prompt}\big(\mathcal{P}_{vp}\big), \quad \mathcal{P}_{vp} \in \mathbb{R}^{H\times W\times 3}, \tag{1}$$

$$\mathcal{V}_{crop} = \texttt{crop_video}\big(\mathcal{P}_{v}, \tau_s, \tau_e\big), \quad \tau_s, \tau_e \in \mathbb{R}^{+},\ \tau_s < \tau_e, \tag{2}$$

where $\mathcal{P}_{vp}$ denotes the visual prompt frame path and $\mathcal{I}_{vp}$ represents the decoded image; $\mathcal{P}_v$ denotes the video path, and τ_s, τ_e denote the start and end timestamps, respectively, yielding the cropped temporal segment $\mathcal{V}_{crop}$. In each round t (where $t = 0, 1, 2, \ldots, T_{\max}$), the model samples a response $\mathcal{R}_t \sim \pi_\theta(\cdot \mid \mathcal{M})$ from the current message context $\mathcal{M}$, which may contain ⟨tool_call⟩ blocks, ⟨answer⟩ blocks, or both. When the model decides to invoke a perception tool, the tool is executed and its result is appended to $\mathcal{M}$ for the next round; when an answer block appears, the `ExtractAnswer` function is called to extract answer A, and the interaction terminates. This iterative cognitive cycle of "active perception → local zoom → evidence-based reasoning" parallels the human cognitive strategy of "global browsing to local close-reading" when confronting complex visual scenes, thereby circumventing the context loss and evidence obscuration inherent in single-pass compression paradigms.

To better illustrate the overall procedure, the entire rollout process is presented in Algorithm 1.

3.2 DATA CONSTRUCTION

Preliminary Data Curation. To construct large-scale high-quality visual prompt video QA data, we propose a fully automated four-stage pipeline that transforms arbitrary video QA datasets into visual-prompt-dependent QA data without any manual annotation.

$$\mathcal{D}_{final} = \mathcal{G}_4 \circ \mathcal{G}_3 \circ \mathcal{G}_2 \circ \mathcal{G}_1(\mathcal{D}_{raw}), \tag{3}$$

where $\mathcal{G}_1$ to $\mathcal{G}_4$ correspond to Filtering, Verification, Mask Generation, and Rendering, respectively.

(1) Low-cost Text Filtering. Since video tokens are computationally expensive, processing all data with video understanding leads to significant resource waste. We employ GPT-4o (Hurst et al., 2024) to rapidly filter pure text QA pairs, eliminating samples unsuitable for visual prompting and preserving only QA pairs targeting concrete visual entities for the next stage:

$$\mathcal{F}_{filter} : \mathcal{D} \mapsto \{0, 1\}, \quad \mathcal{D}_{filter} = \{d \in \mathcal{D}_{raw} \mid \mathcal{F}_{filter}(d) = 1\}, \tag{4}$$

Algorithm 1 Multi-turn Interactive Inference Process of VLM with Environment

Require: Query Q, Visual Prompt Frame $\mathcal{F}_{vp}$, Video $\mathcal{V}$, Tool Set $\mathcal{T} = \{view_visual_prompt,\ crop_video\}$, Policy Model π_θ, Maximum Tool Rounds $T_{\max}$.
Ensure: Final Answer A, Interaction Trajectory $\mathcal{Y}$, Tool Call History $\mathcal{H}$.

Initialization: $\mathcal{Y} \leftarrow \emptyset, t \leftarrow 0, \mathcal{H} \leftarrow \emptyset$.
Encode $\mathcal{V}$ into visual frame sequence: $\mathcal{V}_{frames} \leftarrow \texttt{EncodeVideoFrames}(\mathcal{V})$.
Compose user message $\mathcal{M} \leftarrow \mathcal{V}_{frames} + \{Q,\ \texttt{ToolPrompt}(\mathcal{T},\ \mathcal{F}_{vp})\}$.
while $t \leq T_{\max}$ **do**
  Sample model response: $\mathcal{R}_t \sim \pi_\theta(\cdot \mid \mathcal{M})$.
  Append $\mathcal{R}_t$ to trajectory: $\mathcal{Y} \leftarrow \mathcal{Y} + \mathcal{R}_t$.
  if <tool_call></tool_call> detected in $\mathcal{R}_t$ **then**
    Parse $\{(func_k,\ args_k)\}$ from $\mathcal{R}_t$, append to $\mathcal{H}$.
    Execute tools and append results to $\mathcal{M}$: $\mathcal{M} \leftarrow \mathcal{M} + \texttt{ExecuteTools}(\{(func_k,\ args_k)\})$.
  end if
  if <answer></answer> detected in $\mathcal{R}_t$ **then**
    Extract answer $A \leftarrow \texttt{ExtractAnswer}(\mathcal{R}_t)$.
    return $(A,\ \mathcal{Y},\ \mathcal{H})$.
  end if
  $t \leftarrow t + 1$.
end while
return $(\texttt{NULL},\ \mathcal{Y},\ \mathcal{H})$.

where $\mathcal{D}$ denotes the dataset space and $d = (v, q, a) \in \mathcal{D}$ contains video v, question q, and answer a.

(2) Video-level Verification. For pre-filtered samples, we further verify whether the target is uniquely identifiable in the video. We use Gemini-3.1-Pro (Comanici et al., 2025) to jointly process videos and original QA pairs through a five-step reasoning pipeline: target extraction with uniqueness judgment, generation of a unique semantic tag for SAM3 segmentation, temporal window localization, and QA rewriting with a unified $< vp >$ placeholder:

$$\mathcal{R}_{rewrite} : \mathcal{V} \times \mathcal{QA} \mapsto \mathcal{QA}_{vp}, \quad \mathcal{QA}_{vp} = \mathcal{R}_{rewrite}(\mathcal{V}, \mathcal{QA}; \phi), \tag{5}$$

where ϕ denotes the internal five-step reasoning process comprising target extraction with uniqueness judgment, semantic tag generation for SAM3, temporal window localization, and $< vp >$ substitution.

(3) Pixel-level Mask Generation. Semantic tags alone are insufficient for pixel-level visual prompt rendering. We adopt SAM3 (Carion et al., 2025) to conduct text-driven video diffusion segmentation based on semantic tags, sampling at one frame per second to generate precise pixel-level masks:

$$\mathcal{M}_\tau = \text{SAM3}(\mathcal{V}, \tau; \omega), \quad \forall \tau \in \mathbb{T}, \quad \mathbb{T} = \{\lfloor t \rfloor \mid t \in [0, T)\}, \tag{6}$$

where ω denotes the semantic tag condition and T denotes the total video duration in seconds.

(4) Visual Prompt Rendering. To enhance data diversity and establish alignment between visual prompt symbols and natural language descriptions, we uniformly sample eight visual prompt types and render them on video frames. We then invoke a language model to replace the $< vp >$ placeholder with natural language descriptions corresponding to the visual prompt types, producing visual prompt QA data ready for training:

$$\mathcal{QA}_{rendered} = \texttt{LLM}\big(\mathcal{QA}_{vp}, \mathcal{VP}\big), \tag{7}$$

where $\mathcal{VP}$ denotes the sampled visual prompt type. The unified $< vp >$ facilitates community extensions by enabling seamless substitution across different visual prompt types without modifying downstream model interfaces.

SFT and RL Data Curation. Due to the limited capability of the base VLM, which exhibits poor instruction-following and high tool-calling error rates, we adopt a reject sampling strategy to generate high-quality multi-turn tool-calling trajectories. Specifically, we use data from the Preliminary Data Curation stage as input, and leverage Qwen3-VL-235B-A22B-Thinking to interact with the video environment using predefined tools. Subsequently, a rule-based discriminator filters out trajectories where the model responds incorrectly, ultimately yielding 34.2k high-quality samples for SFT stage. During the RL training phase, we further filter the SFT data based on the pass-k metric, resulting in 4.1k samples for GRPO training.

3.3 TRAINING STRATEGY

Supervised Fine-Tuning. We first conduct SFT to equip the model with foundational behaviors required for multimodal tool-calling VLMs, thereby ensuring effective interaction with the environment. Following the procedure described in Section 3.2, we collect 34.2k high-quality trajectories for training. The model is trained by minimizing the standard autoregressive cross-entropy loss. The objective of SFT is to guide the model toward learning multi-turn, multi-scale active perception patterns in video environments, integrating visual evidence during reasoning, endowing the policy model with basic capabilities for interacting with the video environment, and establishing a foundation for agentic reinforcement learning.

Agentic Reinforcement Learning. In this stage, we treat the model as an agent capable of autonomously using tools, which actively decides whether to view the visual prompt, how to crop segments, and how to integrate retrieved evidence into the reasoning process. We employ GRPO to achieve this objective. The policy model is optimized by maximizing the following objective:

$$\mathbb{E}_{x,\,\{y_i\}_{i=1}^{G}}\left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{\sum_t I(y_{i,t})}\sum_{t:I(y_{i,t})=1}\min\big(r_{i,t},\ \mathrm{clipped}(r_{i,t})\big)\,\hat{A}_{i,t}\right]-\beta\,D_{\mathrm{KL}}(\pi_\theta\|\pi_{\mathrm{ref}}),\quad(8)$$

where $r_{i,t}=\pi_\theta(y_{i,t}|x,y_{i,<t})/\pi_{\mathrm{old}}(y_{i,t}|x,y_{i,<t})$ and $\mathrm{clipped}(r)=\mathrm{clip}(r,1-\epsilon,1+\epsilon)$. The rollout module samples a group of trajectories $\{y_1,y_2,\ldots,y_G\}$ from the old policy π_{old} for each input question x through interaction with the external environment $\mathcal{V}$. The advantage term $\hat{A}_{i,t}$ is computed based on the relative rewards of outputs within each group. Additionally, we introduce a three-component reward modeling approach that jointly optimizes sampled trajectories across three dimensions: answer accuracy, format compliance, and generation efficiency. This design enhances final answer correctness, promotes more effective tool usage during inference, and produces more reliable and well-reasoned trajectories.

1. Answer Accuracy. For the k-th rollout, let $\hat{a}^{(k)}$ and $a^\star$ denote the extracted answer and the ground truth, respectively. We adopt Qwen3-VL-235B-A22B-Instruct (Bai et al., 2025) as a judge to assess their semantic consistency and output a score in $\{1, 0.5, 0\}$ (fully correct, partially correct, or incorrect). The accuracy reward is defined as:

$$R_{acc}^{(k)}=\mathrm{Judge}_{LLM}\big(\hat{a}^{(k)},\,a^\star\big)\ \in\ \{1,\ 0.5,\ 0\}.\quad(9)$$

2. Format Compliance. Let $y^{(k)}$ denote the complete textual output of the k-th rollout and $\mathcal{S}$ be the predefined output schema. This reward encourages the model to consistently produce well-structured outputs with properly organized tool invocations and final answers, enabling reliable downstream parsing and verification. The format reward is computed as:

$$R_{format}^{(k)}=\mathbb{1}\big(y^{(k)}\ \text{matches}\ \mathcal{S}\big).\quad(10)$$

3. Parsimony Reward. We introduce a parsimony reward to encourage the model to accomplish tasks with fewer tool-calling rounds while maintaining answer correctness. Specifically, let $N^{(k)}$ denote the total number of perception tool invocations triggered in the k-th rollout. The parsimony reward is computed as:

$$R_{par}^{(k)}=\max\{0,\ 1-\lambda\cdot N^{(k)}\},\quad(11)$$

where λ controls the strength of the parsimony penalty. This design implicitly incentivizes the model to only invoke tools when additional evidence is needed, thereby achieving a balance between effective reasoning and resource efficiency.

4. Integrated Reward Function. The final reward function is a weighted combination of the three components described above, with weights used to balance the contributions of each component:

$$R^{(k)}=\alpha\cdot R_{acc}^{(k)}+\beta\cdot R_{format}^{(k)}+\gamma\cdot R_{par}^{(k)}.\quad(12)$$

where $\alpha+\beta+\gamma=1$. By integrating these three components into the reward function, our VideoSeeker provides a comprehensive and fine-grained evaluation mechanism, guiding the model to better align with real-world application requirements when optimizing its reasoning capabilities.

Table 1: **Evaluation Results on V2P-Bench across Dimensions.** The "Agent" column indicates whether native tool calling is enabled (✓) or disabled (✗) in the prompt. The best results are **bold** and the second-best are underlined.

Model	Agent	OA	HA	OD	FM	CR	PU	CI	FT	RT	AS	SR	GC	Avg.
Proprietary VLMs														
GPT-4o	✗	76.6	68.9	41.3	60.8	67.0	73.3	**67.6**	**68.1**	70.5	50.0	54.0	48.4	65.4
GPT-4o	✓	75.9	69.6	42.4	58.1	64.8	71.0	65.8	66.4	66.9	48.0	50.8	44.0	63.9
Gemini-2.5-Pro	✗	84.0	72.4	**68.2**	71.8	**75.0**	73.3	22.6	66.7	**72.7**	47.4	67.5	63.6	69.8
Gemini-2.5-Pro	✓	85.2	73.8	67.0	70.5	73.5	71.5	21.0	64.8	71.9	41.0	64.0	60.4	68.5
Open-source VLMs														
LLaVA-OV-7B	✗	57.1	52.1	28.3	47.1	63.8	59.1	41.0	42.1	35.6	63.2	62.8	43.2	52.8
LLaVA-OV-72B	✗	65.5	59.9	34.7	47.0	63.8	43.2	38.5	50.0	41.1	66.3	66.9	45.9	56.7
InternVL3-8B	✗	73.9	69.1	39.1	60.8	58.1	65.9	41.0	52.6	41.1	61.1	69.7	64.9	61.7
LLaVA-Video-7B	✗	60.5	58.1	37.0	49.0	62.9	54.5	41.0	52.6	48.9	57.9	56.6	40.5	54.8
LLaVA-Video-72B	✗	62.2	60.8	30.4	54.9	61.0	54.5	43.6	47.4	42.2	70.5	71.0	59.5	58.6
VideoChat-A1	✓	66.8	60.7	37.9	55.4	43.1	55.3	50.8	42.6	39.8	57.4	61.2	45.9	56.2
REVISOR	✓	70.1	63.9	40.5	58.7	45.8	58.2	53.6	45.1	42.9	60.3	64.1	48.7	58.9
video-o3	✓	72.4	65.8	42.1	60.3	47.2	60.5	55.8	46.9	44.7	62.1	66.3	50.2	60.1
Ours														
Qwen3-VL-4B	✗	73.7	67.1	41.2	58.8	51.7	52.9	51.2	42.1	42.2	63.2	71.9	48.6	59.2
Qwen3-VL-4B	✓	73.0	68.6	26.7	52.9	26.9	62.5	41.0	50.0	44.5	67.6	64.8	40.5	57.6
VideoSeeker-4B	✓	85.7	**79.4**	60.3	70.6	55.6	59.1	61.4	55.6	56.0	76.5	69.2	66.7	70.6
Δ	-	**+12.0**	**+12.3**	**+19.1**	**+11.8**	**+3.9**	**+6.2**	**+10.2**	**+13.5**	**+13.8**	**+13.3**	**-2.7**	**+18.1**	**+11.4**
Qwen3-VL-8B	✗	81.8	68.1	41.3	67.8	45.4	58.3	58.3	47.4	38.5	56.2	70.5	51.3	60.8
Qwen3-VL-8B	✓	71.1	71.2	39.1	58.8	48.3	41.2	64.3	42.1	51.7	57.9	68.4	59.4	59.9
VideoSeeker-8B	✓	**94.3**	79.1	53.7	**76.5**	56.5	**78.6**	66.8	62.3	55.0	**78.8**	**78.8**	**70.3**	**74.5**
Δ	-	**+12.5**	**+11.0**	**+12.4**	**+8.7**	**+11.1**	**+20.3**	**+8.5**	**+14.9**	**+16.5**	**+22.6**	**+8.3**	**+19.0**	**+13.7**

4 EXPERIMENTS

4.1 IMPLEMENTATION DETAILS

Training and Evaluation Setup. In the SFT and RL stages, we leverage 34.2k trajectories and a curated dataset of 4.1k samples collected in Section 3.2. All experiments are built upon Qwen3-VL-4B and Qwen3-VL-8B as base models. We evaluate VideoSeeker against three groups of baselines: open-source models (Video-R1 (Feng et al., 2025), VideoRFT (Wang et al., 2025c), Video-Thinker (Wang et al., 2025d)), proprietary models (GPT-4o (Hurst et al., 2024), Gemini-2.5-Pro (Comanici et al., 2025)), and recent native video-agent methods (REVISOR (Li et al., 2026), video-o3 (Zeng et al., 2026), VideoChat-A1 (Wang et al., 2026)). Evaluations are conducted on V2P-Bench (Zhao et al., 2025b) (a dedicated instance-level benchmark) and three general video understanding benchmarks: Video-MME (Fu et al., 2025), LongVideoBench (Wu et al., 2024), and LongVT (Yang et al., 2025b). Models are deployed on vLLM (Kwon et al., 2023) with OpenAI-SDK-compatible tool calling. All evaluations use temperature 0.

Table 2: **Evaluation Results on General Benchmarks.** The bests are **bold** and the second-best are underlined.

Model	Agent	Video-MME	LongVideoBench	LongVT	Avg.
Proprietary VLMs					
GPT-4o	✗	77.2	**81.3**	17.4	58.6
Gemini-2.5 Pro	✗	**84.8**	-	-	-
Open-source VLMs					
Video-R1-7B	✗	61.0	-	27.9	-
VideoRFT-7B	✗	60.9	-	26.5	-
Video-Thinker-7B	✗	61.0	-	10.4	-
LongVT-7B	✗	66.1	-	31.0	-
VideoChat-A1	✓	63.5	56.9	33.8	51.4
REVISOR	✓	61.8	60.3	39.5	53.9
video-o3	✓	65.2	58.4	41.0	54.9
Ours					
Qwen3-VL-4B	✗	65.3	62.6	38.5	55.5
Qwen3-VL-4B	✓	61.5	50.4	36.9	49.6
VideoSeeker-4B	✓	66.1	64.2	45.7	58.7
Δ	-	**+0.8**	**+1.6**	**+7.2**	**+3.2**
Qwen3-VL-8B	✗	67.4	64.6	39.4	57.1
Qwen3-VL-8B	✓	58.3	42.9	11.7	37.6
VideoSeeker-8B	✓	68.1	66.5	**46.5**	**60.4**
Δ	-	**+0.7**	**+1.9**	**+7.1**	**+3.3**

Training Infrastructure. We conduct SFT on LLaMA-Factory (Zheng et al., 2024) and RL training on verl (Sheng et al., 2024), both employing full-parameter fine-tuning. All experiments are performed on 8 NVIDIA H800 GPUs. More detailed training hyperparameters are provided in Appendix B.3.

4.2 Main Results

As illustrated in Table 1, our VideoSeeker series achieves the best performance among open-source models and is competitive with powerful closed-source models. VideoSeeker-4B improves over the Qwen3-VL-4B baseline by +11.4% on average, with particularly notable gains on HA, OD, and AS; scaling to VideoSeeker-8B further improves over Qwen3-VL-8B by +13.7% on average, with clear gains across most fine-grained dimensions, surpassing Gemini-2.5-Pro and GPT-4o. Notably, enabling the same tools for GPT-4o and Gemini-2.5-Pro without training in fact degrades their V2P-Bench averages (65.4 $\rightarrow$ 63.9 and 69.8 $\rightarrow$ 68.5; Table 1), due to insufficient tool instruction-following—incorrect function names, wrong arguments, and inability to recover from errors; these cause deadlock loops that prevent effective answers. The gain is therefore not simply in-domain fine-tuning beating zero-shot baselines. Beyond general VLMs, VideoSeeker also outperforms recent native video-agent methods (REVISOR (Li et al., 2026), video-o3 (Zeng et al., 2026), VideoChat-A1 (Wang et al., 2026)) under the same agent protocol; they reach only 58.9 / 60.1 / 56.2 on average, as their text-query-grounding or holistic-reasoning origins lead them to crop temporally before localizing the target on the visual-prompt frame, losing it in multi-instance scenes. Moreover, although our training data comes exclusively from instance-level tasks, VideoSeeker transfers to general video understanding benchmarks (Table 2; analysis in §4.4), achieving average improvements of +3.2% and +3.3% over three tasks.

4.3 Ablation Studies

Tools Ablation. As shown in Table 3, we decomposed the tool set to analyze the contribution of each tool. The consistent performance improvements brought by the gradual introduction of the tool set clearly validate the effectiveness of our methodological paradigm. Notably, **the combination of the two tools yields synergistic gains** that exceed their individual contributions, indicating that the two tools form a complementary relationship in information acquisition. Beyond downstream QA accuracy, we further verify `crop_video`'s localization benefit directly through temporal grounding evaluation on three VTG benchmarks (Appendix C.2).

Table 3: **Tools Ablation.**

VP.	Crop.	Avg.
Qwen3-VL-8B *(Baseline)*		60.8
✓		69.4
	✓	63.7
✓	✓	74.5

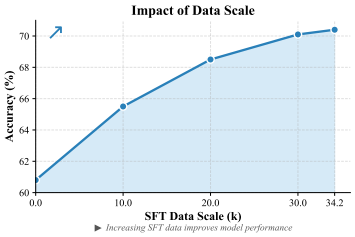

Figure 3: **Effect of Data Scale.**

Data Ablation. We construct several subsets by progressively increasing the sampling ratio from the full training corpus to investigate the impact of SFT data scale on model performance. As shown in Figure 3, performance improves with increasing data volume, and the gains gradually diminish as the data scale expands further. This observation reveals a prominent **diminishing marginal returns pattern** in performance improvement, where the model approaches saturation beyond a certain data scale. These findings provide insights for balancing dataset scale and model performance.

Reward Ablation. As shown in Table 4, **our reward system provides a stable training signal**, and we systematically analyze the contribution of each reward signal during RL training. The format reward substantially outperforms the accuracy-only baseline, while the efficiency reward encourages more concise tool usage. Notably, the combined three-reward scheme surpasses the sum of individual contributions, revealing complementary effects across reward dimensions that jointly enhance effective reasoning.

Table 4: **Reward Ablation.**

Reward Type R_{acc}	Reward Type R_{format}	Reward Type R_{eff}	Acc.
✓			65.4
✓	✓		73.1
✓		✓	68.7
✓	✓	✓	74.5

Table 5: **Stage Ablation.**

SFT	RL (Single Turn)	RL (Agentic)	Acc.
Qwen3-VL-8B *(Baseline)*			60.8
✓			70.4
	✓		62.6
		✓	65.9
✓		✓	74.5

Stage Ablation. As shown in Table 5, we systematically ablate the contributions of the SFT and RL training stages to model performance. Experimental results demonstrate that high-quality SFT data endows the model with **robust reasoning patterns**, yielding a substantial performance boost (+9.6%). In the zero-shot RL setting, single-turn RL leads to marginal improvement (+1.8%). In contrast, agentic RL paradigm achieves +5.1% improvement, which is more ef-

fective (+3.3%) than single-turn RL. **This validates the agentic paradigm as a critical enabler for effective RL training** on instance-level video understanding tasks. The cascaded two-stage training paradigm leverages synergistic gains from both strategies, achieving optimal performance (74.5%) and thereby establishing the optimal training pipeline in our framework.

4.4 Analysis

Generalization to General Video Understanding Tasks. Despite being trained exclusively on instance-level tasks, VideoSeeker demonstrates strong cross-task generalization on general video benchmarks (Table 2), achieving +3.2% and +3.3% average improvements. This reveals that core capabilities learned from instance-level tasks, such as long-range visual reasoning and multi-turn tool-augmented reasoning, transfer compositionally to broader scenarios, which highlights the value of instance-level data in instilling generalizable priors without additional general data.

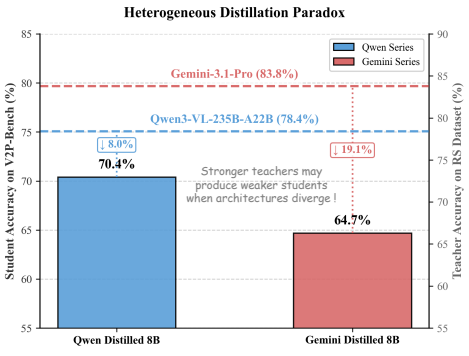

Figure 4: **Distillation Paradox.**

The heterogeneous distillation paradox: stronger teachers may produce weaker students. We experiment with two teacher models: Qwen3-VL-235B-A22B-Thinking and Gemini-3.1-Pro, achieving 78.4% and 83.8% accuracy on the rejection-sampled dataset respectively. After SFT training Qwen3-VL-8B, the resulting student models achieve 70.4% and 64.7% on V2P-Bench, with relative performance degradation of 8.0% and 19.1%. As illustrated in Figure 4, this reveals a counter-intuitive finding: **The raw capability of a teacher model does not proportionally transfer to distillation performance**. In homogeneous distillation, teachers and students share similar patterns, enabling efficient knowledge transfer; in heterogeneous distillation, pattern divergence is significant, causing stronger teachers' knowledge to be less effectively absorbed and leading to greater performance degradation.

Pareto Frontier. We jointly report accuracy and true inference time (wall-clock) by adding a non-agentic single-pass baseline (no tool use) alongside the untrained agent and VideoSeeker (Figure 5). VideoSeeker strictly dominates the untrained agent: on both 4B and 8B it is simultaneously more accurate and faster, so the untrained agent is a strictly dominated solution off the Pareto frontier. VideoSeeker and single-pass inference jointly lie on the accuracy–time Pareto frontier: single-pass is fastest but low-accuracy, while VideoSeeker trades acceptable extra time for a +11 to +14 accuracy gain.

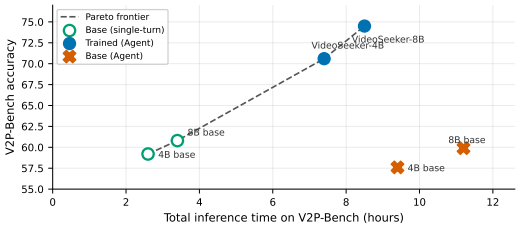

Figure 5: **Pareto frontier.**

Case Study. Appendix D.1 shows how VideoSeeker invokes tools to examine instance targets, clips specific segments for precise localization, and completes the task. This agentic paradigm enables high-precision instance-level video understanding, avoiding the localization errors of methods that rely on vague textual descriptions.

5 Conclusion

In this work, we propose VideoSeeker, an agentic paradigm that enables LVLMs to perform instance-level video understanding through native tool invocation. By integrating agentic reasoning with instance-level video understanding tasks, VideoSeeker proactively perceives and retrieves relevant video segments on demand, achieving more precise spatial and temporal references than text-only approaches. We construct a four-stage fully automated data synthesis pipeline generating large-scale instance-level data, and develop a two-stage training strategy to internalize tool-calling capabilities into LVLMs. Experiments on V2P-Bench show VideoSeeker-8B gains 13.7% on average, surpassing GPT-4o and Gemini-2.5-Pro, while transferring effectively to broader video understanding scenarios.

AI Assistance Statement

Large language models were used to assist with the writing and revision of this paper, including grammar correction, improving clarity of expression, text organization, and LaTeX formatting. The research ideas and experimental design were proposed and completed by the authors. The authors made the final decisions on the paper's content and take responsibility for its scientific content, experimental results, and conclusions.

Ethics Statement

We adhere to the ICLR 2027 Code of Ethics throughout this research. This work uses only public academic datasets under their respective licenses, with human-drawn annotations and no sensitive or private content. The training data is entirely open-source and does not reflect social or domain bias. The study does not involve human subjects or any sensitive personal information.

Reproducibility Statement

Details supporting reproducibility, including training hyperparameters, the data construction pipeline, and the evaluation protocol, are provided in the main text and the appendix. All base models and training frameworks used in this work are publicly available. Upon acceptance, we will release the training and evaluation code along with the constructed dataset.

References

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719*, 2025.

Aili Chen, Aonian Li, Bangwei Gong, Binyang Jiang, Bo Fei, Bo Yang, Boji Shan, Changqing Yu, Chao Wang, Cheng Zhu, et al. Minimax-m1: Scaling test-time compute efficiently with lightning attention. *arXiv preprint arXiv:2506.13585*, 2025a.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. In *European Conference on Computer Vision*, pp. 370–387. Springer, 2024a.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? *Advances in Neural Information Processing Systems*, 37:27056–27087, 2024b.

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*, 2025b.

Christopher Clark, Jieyu Zhang, Zixian Ma, Jae Sung Park, Mohammadreza Salehi, Rohun Tripathi, Sangho Lee, Zhongzheng Ren, Chris Dongjoo Kim, Yinuo Yang, et al. Molmo2: Open weights and data for vision-language models with video understanding and grounding. *arXiv preprint arXiv:2601.10611*, 2026.

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.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 91–104, 2025.

Yihe Deng, Hritik Bansal, Fan Yin, Nanyun Peng, Wei Wang, and Kai-Wei Chang. Openvlthinker: Complex vision-language reasoning via iterative sft-rl cycles. *arXiv preprint arXiv:2503.17352*, 2025.

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.

Chaoyou Fu, Yuhan Dai, Yongdong 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. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 24108–24118, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Jack Hong, Chenxiao Zhao, ChengLin Zhu, Weiheng Lu, Guohai Xu, and Xing Yu. Deepeyesv2: Toward agentic multimodal model. *arXiv preprint arXiv:2511.05271*, 2025.

Wenyi Hong, Xiaotao Gu, Ziyang Pan, Zhen Yang, Yuting Wang, Yue Wang, Yuanchang Yue, Yu Wang, Yanling Wang, Yan Wang, et al. Glm-5v-turbo: Toward a native foundation model for multimodal agents. *arXiv preprint arXiv:2604.26752*, 2026.

Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Xu Tang, Yao Hu, and Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models. *arXiv preprint arXiv:2503.06749*, 2025.

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.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pp. 611–626, 2023.

Jiaze Li, Hao Yin, Wenhui Tan, Jingyang Chen, Boshen Xu, Yuxun Qu, Yijing Chen, Jianzhong Ju, Zhenbo Luo, and Jian Luan. Revisor: Beyond textual reflection, towards multimodal introspective reasoning in long-form video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5059–5069, June 2026.

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.

Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi Wang. Visual-rft: Visual reinforcement fine-tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2034–2044, 2025.

Fanqing Meng, Lingxiao Du, Zongkai Liu, Zhixiang Zhou, Quanfeng Lu, Daocheng Fu, Tiancheng Han, Botian Shi, Wenhai Wang, Junjun He, et al. Mm-eureka: Exploring the frontiers of multimodal reasoning with rule-based reinforcement learning. *arXiv preprint arXiv:2503.07365*, 2025.

Yukun Qi, Yiming Zhao, Yu Zeng, Xikun Bao, Wenxuan Huang, Lin Chen, Zehui Chen, Jie Zhao, Zhongang Qi, and Feng Zhao. Vcr-bench: A comprehensive evaluation framework for video chain-of-thought reasoning. *arXiv preprint arXiv:2504.07956*, 2025.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

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

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Haozhan Shen, Peng Liu, Jingcheng Li, Chunxin Fang, Yibo Ma, Jiajia Liao, Qiaoli Shen, Zilun Zhang, Kangjia Zhao, Qianqian Zhang, et al. Vlm-r1: A stable and generalizable r1-style large vision-language model. *arXiv preprint arXiv:2504.07615*, 2025.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*, 2024.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, SH Cai, Yuan Cao, Y Charles, HS Che, Cheng Chen, Guanduo Chen, et al. Kimi k2. 5: Visual agentic intelligence. *arXiv preprint arXiv:2602.02276*, 2026.

OpenAI Team. Thinking with images. `https://openai.com/index/thinking-with-images/`, 2025.

Shulin Tian, Ruiqi Wang, Hongming Guo, Penghao Wu, Yuhao Dong, Xiuying Wang, Jingkang Yang, Hao Zhang, Hongyuan Zhu, and Ziwei Liu. Ego-r1: Chain-of-tool-thought for ultra-long egocentric video reasoning. *arXiv preprint arXiv:2506.13654*, 2025.

Chaoyang Wang, Kaituo Feng, Dongyang Chen, Zhongyu Wang, Zhixun Li, Sicheng Gao, Meng Meng, Xu Zhou, Manyuan Zhang, Yuzhang Shang, et al. Adatooler-v: Adaptive tool-use for images and videos. *arXiv preprint arXiv:2512.16918*, 2025a.

Haozhe Wang, Alex Su, Weiming Ren, Fangzhen Lin, and Wenhu Chen. Pixel reasoner: Incentivizing pixel-space reasoning with curiosity-driven reinforcement learning. *arXiv preprint arXiv:2505.15966*, 2025b.

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*, 2025c.

Shijian Wang, Jiarui Jin, Xingjian Wang, Linxin Song, Runhao Fu, Hecheng Wang, Zongyuan Ge, Yuan Lu, and Xuelian Cheng. Video-thinker: Sparking" thinking with videos" via reinforcement learning. *arXiv preprint arXiv:2510.23473*, 2025d.

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*, 2025e.

Ye Wang, Ziheng Wang, Boshen Xu, Yang Du, Kejun Lin, Zihan Xiao, Zihao Yue, Jianzhong Ju, Liang Zhang, Dingyi Yang, et al. Time-r1: Post-training large vision language model for temporal video grounding. *arXiv preprint arXiv:2503.13377*, 2025f.

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. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 10467–10475, 2026.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. Longvideobench: A benchmark for long-context interleaved video-language understanding. *Advances in Neural Information Processing Systems*, 37:28828–28857, 2024.

Junfei Wu, Jian Guan, Kaituo Feng, Qiang Liu, Shu Wu, Liang Wang, Wei Wu, and Tieniu Tan. Reinforcing spatial reasoning in vision-language models with interwoven thinking and visual drawing. *arXiv preprint arXiv:2506.09965*, 2025.

Long Xing, Xiaoyi Dong, Yuhang Zang, Yuhang Cao, Jianze Liang, Qidong Huang, Jiaqi Wang, Feng Wu, and Dahua Lin. Caprl: Stimulating dense image caption capabilities via reinforcement learning. *arXiv preprint arXiv:2509.22647*, 2025.

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. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2087–2098, 2025.

Yi Yang, Xiaoxuan He, Hongkun Pan, Xiyan Jiang, Yan Deng, Xingtao Yang, Haoyu Lu, Dacheng Yin, Fengyun Rao, Minfeng Zhu, et al. R1-onevision: Advancing generalized multimodal reasoning through cross-modal formalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2376–2385, 2025a.

Zuhao Yang, Sudong Wang, Kaichen Zhang, Keming Wu, Sicong Leng, Yifan Zhang, Bo Li, Chengwei Qin, Shijian Lu, Xingxuan Li, et al. Longvt: Incentivizing" thinking with long videos" via native tool calling. *arXiv preprint arXiv:2511.20785*, 2025b.

En Yu, Kangheng Lin, Liang Zhao, Jisheng Yin, Yana Wei, Yuang Peng, Haoran Wei, Jianjian Sun, Chunrui Han, Zheng Ge, et al. Perception-r1: Pioneering perception policy with reinforcement learning. *arXiv preprint arXiv:2504.07954*, 2025.

Xiangyu Zeng, Zhiqiu Zhang, Yuhan Zhu, Xinhao Li, Zikang Wang, Changlian Ma, Qingyu Zhang, Zizheng Huang, Kun Ouyang, Tianxiang Jiang, et al. Video-o3: Native interleaved clue seeking for long video multi-hop reasoning. *arXiv preprint arXiv:2601.23224*, 2026.

Yu Zeng, Wenxuan Huang, Shiting Huang, Xikun Bao, Yukun Qi, Yiming Zhao, Qiuchen Wang, Lin Chen, Zehui Chen, Huaian Chen, et al. Agentic jigsaw interaction learning for enhancing visual perception and reasoning in vision-language models. *arXiv preprint arXiv:2510.01304*, 2025a.

Yu Zeng, Yukun Qi, Yiming Zhao, Xikun Bao, Lin Chen, Zehui Chen, Shiting Huang, Jie Zhao, and Feng Zhao. Enhancing large vision-language models with ultra-detailed image caption generation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 26703–26729, 2025b.

Haoji Zhang, Xin Gu, Jiawen Li, Chixiang Ma, Sule Bai, Chubin Zhang, Bowen Zhang, Zhichao Zhou, Dongliang He, and Yansong Tang. Thinking with videos: Multimodal tool-augmented reinforcement learning for long video reasoning. *arXiv preprint arXiv:2508.04416*, 2025.

Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Llava-video: Video instruction tuning with synthetic data. *arXiv preprint arXiv:2410.02713*, 2024.

Shitian Zhao, Haoquan Zhang, Shaoheng Lin, Ming Li, Qilong Wu, Kaipeng Zhang, and Chen Wei. Pyvision: Agentic vision with dynamic tooling. *arXiv preprint arXiv:2507.07998*, 2025a.

Yiming Zhao, Yu Zeng, Yukun Qi, YaoYang Liu, Xikun Bao, Lin Chen, Zehui Chen, Qing Miao, Chenxi Liu, Jie Zhao, et al. V2p-bench: Evaluating video-language understanding with visual prompts for better human-model interaction. *arXiv preprint arXiv:2503.17736*, 2025b.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 3: system demonstrations)*, pp. 400–410, 2024.

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. *arXiv preprint arXiv:2505.14362*, 2025.

SUPPLEMENTARY MATERIAL

A Task and Contribution 15

A.1 Clarification of Tool Contribution 15

A.2 Applications of Instance-Level Video Understanding 15

A.3 Task Novelty 15

B Data, Benchmarks, and Implementation 16

B.1 Dataset Details 16

B.2 Benchmark Information 17

B.3 Hyperparameters 17

C Additional Evaluation and Analysis 17

C.1 Additional Ablation 17

C.2 Temporal Grounding Evaluation 18

C.3 Generalization Analysis 19

C.4 Cross-Backbone Generalization 19

D Case Studies and Prompts 19

D.1 Case Study 19

D.2 Prompts 22

A Task and Contribution

A.1 Clarification of Tool Contribution

We do not claim the two tools themselves as our contribution. `crop_video` is a common, well-established tool in mainstream "thinking with videos" work (REVISOR, video-o3); `view_visual_prompt` is a minimal, generic "fetch-a-reference-image-by-path" primitive with no task-specific machinery, and it integrates into any unified toolkit. We are therefore no less general than mainstream designs: both tools are standard primitives.

A "generic toolkit," however, does not equate to "the problem being solved." When we hand the toolkit directly to untrained open- and closed-source models, their V2P-Bench average drops for all four models (Table 6):

Table 6: **Training-free tool exposure degrades all base models.**

Model	Agent	V2P-Bench Avg.	Δ
Qwen3-VL-4B	✗	59.2	–
Qwen3-VL-4B	✓	57.6	−1.6
Qwen3-VL-8B	✗	60.8	–
Qwen3-VL-8B	✓	59.9	−0.9
GPT-4o	✗	65.4	–
GPT-4o	✓	63.9	−1.5
Gemini-2.5-Pro	✗	69.8	–
Gemini-2.5-Pro	✓	68.5	−1.3

All four models degrade uniformly when given the tools without training. Availability of tools is not the bottleneck; learning to use them is. The agentic training paradigm that teaches a model to use tools is the contribution of this work.

For the toolset, we use only two standard primitives (fetch-image-by-path and temporal crop) rather than a large "unified spatio-temporal grounding toolkit." The design is minimal and tailored to the practical needs of video visual-prompt human–model interaction: fewer, more standard tools yield a smaller tool-calling error surface and are easier for any harness to adopt. This is not a deficiency in generality.

A.2 Applications of Instance-Level Video Understanding

This interaction paradigm has broad practical value in deployment:

Interaction with Mobile Devices. Users can watch videos on mobile devices such as smartphones and computers, easily creating visual prompts with their fingers or a mouse. This functionality has been successfully implemented in systems like GPT-4o and Video Refer, allowing users to interact with video content more intuitively, enhancing their viewing experience.

Smart Device Wearables. Imagine users wearing advanced smart devices (like Apple Vision Pro), immersed in a sea of video content. These users could generate visual prompts through natural gestures. It undoubtedly offers limitless possibilities for future interaction methods, significantly enhancing the immersive experience of watching videos.

In these diverse scenarios, users can interact smoothly with local or cloud-based LVLMs based on video content. Developers only need to adjust the front-end system to efficiently capture video frames and visual prompts to achieve this goal. Such integration will create a more immersive experience for users, transforming video watching from passive observation into an engaging and interactive journey.

A.3 Task Novelty

We delineate the novelty of our task formulation by comparing it with related areas along four key dimensions (Table 7):

Table 7: **Comparison with related areas along four dimensions.**

Area	Referent	Instance-level	Open-ended QA	Agentic
Visual grounding	text / image	✓	✗	✗
Referring VOS	text	✓	✗	✗
Interactive video QA	text	✗	✓	✗
Video seg./tracking	class / click	✓	✗	✗
Tool-augmented video reasoning	text query	✗	✓	✓
Ours (VideoSeeker)	visual prompt	✓	✓	✓

No single related area simultaneously possesses the combination of "visual-prompt referent + instance-level + open-ended QA + agentic active perception": the related areas either refer via text, or output only boxes/masks/tracks, or address only global semantics—our setting falls precisely outside their intersection. The novelty is therefore not a simple combination of existing ideas; the task fundamentally requires the model to (1) associate the visual prompt (rather than text) with the correct target instance, (2) reason over the instance's temporal dynamics across the whole video, and (3) produce open-ended answers. This combination of "visual-prompt referent + instance-level + open-ended video reasoning" has not been systematically studied by any single prior area. This setting directly serves real-world video human–model interaction scenarios (see Appendix A.2).

B Data, Benchmarks, and Implementation

B.1 Dataset Details

Data Source. Our data construction pipeline uses the original videos and QA data from LLaVA-Video-178K (Zhang et al., 2024) as source material, comprising 178k videos and approximately 1.3 million instruction samples. This dataset integrates 10 mainstream video sources, covering domains such as activity recording, cooking, film, first-person perspective, and more. Multi-dimensional filtering rules are applied to select unedited raw videos with rich temporal variations, ensuring narrative completeness. Figure 8 illustrates the source distribution of the video data.

Table 8: **Video source distribution.**

Source	Count	Ratio
YouTube	5,405	72.1%
Charades	826	11.0%
ActivityNet	526	7.0%
YouCook2	454	6.1%
NextQA	258	3.4%
Ego4D	28	0.4%

Data Construction Pipeline. As described in Section 3.2, we propose a fully automated four-stage pipeline, starting from 147,245 raw video QA samples from LLaVA-Video-178K (Zhang et al., 2024) and transforming them into visual-prompt-dependent QA data: *(1) Text Filtering*, using GPT-4o to quickly pre-filter text QA and remove samples unsuitable for visual prompts (e.g., camera/cinematography questions, scene/background descriptions, overall activity summaries, counting questions, abstract/non-visual questions, ambient lighting/color queries), retaining 44.5%; *(2) Video Verification*, using Gemini-3.1-Pro to perform five-step reasoning jointly with the video, excluding multi-target ambiguous samples, retaining 32.9%; *(3) SAM3 Segmentation*, generating pixel-level masks at 1 fps based on semantic labels, retaining 27.9%; *(4) Visual Prompt Rendering*, uniformly sampling 8 visual prompt types (rectangle, mask contour, ellipse, triangle, scribble, point, arrow, set-of-mark) to render on video frames and rewrite QA, retaining 27.8%. Table 9 presents detailed stage and statistics information. The final dataset covers 8 visual prompt types, providing diverse spatial and geometric variations for model training.

Table 9: **Data pipeline retention statistics.**

Stage	Output	Relative Retention	Description
Step 0: Original Dataset	147,245	100.0%	LLaVA-Video-178K original data
Step 1: Text Filtering	65,594	44.5%	GPT-4o filters text QA
Step 2: Video Verification	48,457	32.9%	Gemini-3.1-Pro serves as a verifier
Step 3: SAM3 Segmentation	41,083	27.9%	1 FPS pixel-level mask generation
Step 4: Visual Prompt Rendering	40,929	27.8%	Rendering and QA rewriting

B.2 Benchmark Information

We evaluate on four video understanding benchmarks, including one instance-level video understanding benchmark and three general video understanding benchmarks. This section introduces each benchmark. During evaluation, we uniformly segment videos into 256 frames on average.

- **V2P-Bench (Zhao et al., 2025b)** is a benchmark for evaluating LVLMs on visual-prompt-driven instance-level video understanding. Unlike text-only approaches, it introduces visual prompting to require precise spatial-temporal reasoning. It contains 980 videos with 1,172 QA pairs, covering three core tasks across twelve evaluation dimensions, assessing instance-level video understanding.
- **Video-MME (Fu et al., 2025)** is a video understanding benchmark for multimodal LLMs, evaluating capabilities in long-video and complex-reasoning scenarios. It contains approximately 900 manually curated videos spanning multiple domains, with 2.7K multiple-choice QA pairs. All data undergo rigorous human annotation. The dataset supports video, subtitles, and audio inputs. In our evaluation, we exclusively use video modality.
- **LongVideoBench (Wu et al., 2024)** is a large-scale benchmark for long-context video-language understanding, evaluating multimodal models on videos up to one hour. It contains 3,763 diverse web videos covering movies, daily life, knowledge, and news, with 6,678 human-annotated multiple-choice questions. Video durations range from 8 seconds to 60 minutes. Its core innovation is the "Referring Reasoning" paradigm, embedding referring queries to locate relevant segments and requiring both precise retrieval and coherent contextual reasoning.
- **LongVT (Yang et al., 2025b)** is a benchmark for long-video open-domain question answering, containing 244 long videos and 1,280 QA pairs verified through manual review. The average video duration is approximately 1,688 seconds, with most videos (71.84%) in the 15-30 minute range and 28.16% exceeding 30 minutes. Its core design features a "needle-in-a-haystack" setting where supporting evidence exists only in narrow time windows, effectively evaluating models' abilities to locate and reason about fine-grained information within long videos.

B.3 Hyperparameters

We detail the hyperparameters used in our training in Table 10 and Table 11. During agentic RL training, we set $\alpha = 0.8$, $\beta = 0.15$, $\gamma = 0.05$. For both SFT and RL, we sample videos at 1 fps: the first round encodes the full video into at most 256 frames, and each `crop_video` call re-samples the cropped temporal window into at most 64 frames, with this per-crop budget independent across rounds. The maximum single-frame resolution is set to 112896. During both SFT and RL training, LLaMA-Factory and verl automatically inject timestamps for videos, while during inference, we manually add corresponding timestamps to each frame.

C Additional Evaluation and Analysis

C.1 Additional Ablation

The SFT data is itself multi-turn: each trajectory contains multiple rounds of `<tool_call>`, the tool-returned evidence, and the final `<answer>`, and the SFT loss covers the entire trajectory. Whether to retain the multi-turn tool-calling structure is therefore a decomposable variable. The main-text stage ablation (Table 5) already shows, under the zero-RL setting, that the agentic variant outperforms the single-turn variant (62.6 $\rightarrow$ 65.9, +3.3).

To separate the agentic paradigm's contribution from in-domain fine-tuning, we distill a single-turn SFT baseline without tool use, denoted SFT (Single Turn), from the same $\sim$41k visual-prompt QA data and under the same setting as SFT (Agentic). Table 12 reports the full single-turn vs. agentic decomposition across both the SFT and RL stages on V2P-Bench (Qwen3-VL-8B).

The multi-turn vs. single-turn gap is consistent across three levels. At the SFT stage, SFT (Single Turn) yields only +4.1 (60.8 $\rightarrow$ 64.9), whereas the same-scale SFT (Agentic) that retains multi-turn

Table 10: Key hyperparameters for SFT.

Name	Value
Finetuning type	Full
Freeze vision tower	True
Freeze multi-modal projector	False
Freeze language model	False
Cutoff len	16384
image/video max pixels	112896
Video FPS	1.0
Video max length	256
Crop max frames (per round)	64
Batch size per device	1
Gradient accumulation steps	2
Learning rate	1.0e-5
LR scheduler type	cosine
Warmup ratio	0.1
Epochs	1.0

Table 11: Key hyperparameters for RL.

Name	Value
Algorithm	GRPO
Max tool rounds	5
Crop max frames (per round)	64
Agent loop	Tool agent
Rollout num	8
Train batch size	32
Mini batch size	8
Micro batch size per GPU	1
Learning rate	1.0e-6
KL loss coefficient	0.001
Entropy coefficient	0.0
Max prompt length	16384
Max response length	4096
Total epochs	1
GPU memory utilization	0.8

Table 12: **Single-turn vs. agentic SFT/RL ablations.**

SFT (ST)	SFT (Agentic)	RL (ST)	RL (Agentic)	Acc.
Qwen3-VL-8B *(Baseline)*				60.8
✓				64.9
	✓			70.4
		✓		62.6
			✓	65.9
✓		✓		67.3
	✓		✓	74.5

tool calling yields +9.6 (60.8 → 70.4), a +5.5 gap. At the zero-RL stage, RL (Single Turn) yields +1.8 (60.8 → 62.6) while RL (Agentic) yields +5.1 (60.8 → 65.9), a +3.3 gap. For the full cascaded pipeline, SFT (Single Turn) as cold-start followed by an equally tool-free RL (Single Turn) reaches 67.3 (+6.5); this is the ceiling of the single-turn paradigm, yet it remains below the 70.4 of SFT (Agentic) alone and the 74.5 of the full method.

The single-turn paradigm's full two-stage gain (+6.5) is thus below the agentic method's SFT-only gain (+9.6), which rules out the alternative explanation of "merely more training stages." The gain comes from the agentic multi-turn tool-calling paradigm itself, not from fine-tuning on in-domain data. Single-turn inference is a form of passive perception in which the model cannot look back at the visual context, whereas the multi-turn agentic paradigm lets the model actively revisit the video content via tool calls. Visual-prompt QA requires the model to precisely localize and finely inspect the visual-prompt frame, for which the multimodal returns of our tools provide strong visual evidence.

C.2 Temporal Grounding Evaluation

To directly evaluate `crop_video`'s temporal localization accuracy, we evaluate VideoSeeker on three classic video temporal grounding (VTG) benchmarks—Charades-STA, ActivityNet Captions, and QVHighlights—using IoU metrics, following the protocol of recent MLLM-based VTG methods. We include the single-turn SFT+RL variant as a control. Table 13 reports the results.

Even without any VTG-specific training, VideoSeeker attains consistent localization-accuracy gains over the base across all three benchmarks (4B: mean mIoU 36.6 → 40.2, +3.6; 8B: 38.7 → 43.0, +4.3). This directly and quantitatively verifies the model's temporal grounding ability, rather than relying solely on downstream QA accuracy. The control is more telling: the single-turn SFT+RL variant falls markedly below the base (4B: 36.6 → 31.9, -4.7; 8B: 38.7 → 34.1, -4.6), showing that

Table 13: **Temporal grounding evaluation on three VTG benchmarks (mIoU).**

	Charades-STA				ActivityNet				QVHighlights				
Model	R@0.3	R@0.5	R@0.7	mIoU	R@0.3	R@0.5	R@0.7	mIoU	R@0.3	R@0.5	R@0.7	mIoU	**Avg.**
Qwen3-VL-4B (Base)	54.2	39.8	19.3	37.1	46.3	29.7	15.2	33.8	55.8	41.2	22.6	38.9	36.6
+ SFT+RL (Single-Turn)	49.1	34.6	15.2	32.4	41.3	24.9	11.4	29.2	50.5	36.1	18.5	34.2	31.9
VideoSeeker-4B (Multi-Turn)	58.9	44.6	23.1	41.2	50.1	33.2	17.9	36.9	60.3	46.1	26.4	42.5	40.2
Qwen3-VL-8B (Base)	57.1	42.5	21.8	39.4	48.9	32.1	16.8	35.6	58.4	44.0	24.9	41.0	38.7
+ SFT+RL (Single-Turn)	52.3	37.6	17.7	35.1	43.7	27.3	12.6	31.0	53.4	38.8	20.2	36.3	34.1
VideoSeeker-8B (Multi-Turn)	62.7	48.3	26.9	44.5	53.4	36.5	20.3	39.2	63.9	49.7	29.2	45.3	43.0

the improvement in localization comes from the multi-turn agentic RL rather than the training per se. This is consistent with the mechanism of `crop_video`: actively cropping and re-inspecting suspicious windows at a higher effective frame rate is the source of improved localization accuracy.

C.3 Generalization Analysis

The gains across the general benchmarks are uneven (Video-MME +0.7, LongVT +7.1), which is a manifestation of our method's mechanism rather than an overstatement. VideoSeeker's core capability is active temporal grounding and fine-grained instance inspection, which only helps when "the question requires localizing a specific instance/moment":

- Video-MME is dominated by holistic/global-semantic questions (e.g., "What is the genre of this video?"), answerable from a single global glance without precise localization, so `crop_video` has little to contribute and the gain is small (+0.7).
- LongVT questions depend heavily on fine-grained temporal grounding (e.g., "After the woman demonstrates the gray cardigan, the video briefly shows only another person's legs. What garment is that person wearing in this cutaway shot?"), with the answer confined to a narrow temporal window—exactly VideoSeeker's target scenario—so the gain is large (+7.1).

Thus the distribution of gains aligns with "whether the question requires localization." This shows the improvement causally stems from our localization mechanism rather than an unexplained generic boost.

C.4 Cross-Backbone Generalization

Our method is backbone-agnostic by design: the pipeline of automatic data synthesis plus SFT/RL internalization of tool calling contains no Qwen-specific assumptions and requires only that the backbone have basic multi-turn function-calling ability. To verify this, we apply the full pipeline to backbones of different families and scales (Table 14):

Table 14: **Cross-backbone Generalization.**

Backbone	**Family / Scale**	**Base (V2P)**	**+ VideoSeeker**	Δ
Qwen3-VL-8B	Qwen / 8B	60.8	74.5	+13.7
InternVL3.5-8B	InternVL / 8B	57.8	66.4	+8.6
MiMo-VL-7B	MiMo / 7B	55.6	66.2	+10.6

Across three different families and scales from 4B to 8B, VideoSeeker attains consistent and significant gains (+8.6 to +13.7), showing that the method's benefit does not depend on a particular backbone or scale. Combined with the Qwen3-VL-4B result in the main text (+11.4), the method is effective across backbones and scales.

D Case Studies and Prompts

D.1 Case Study

See Figure 6 and Figure 7.

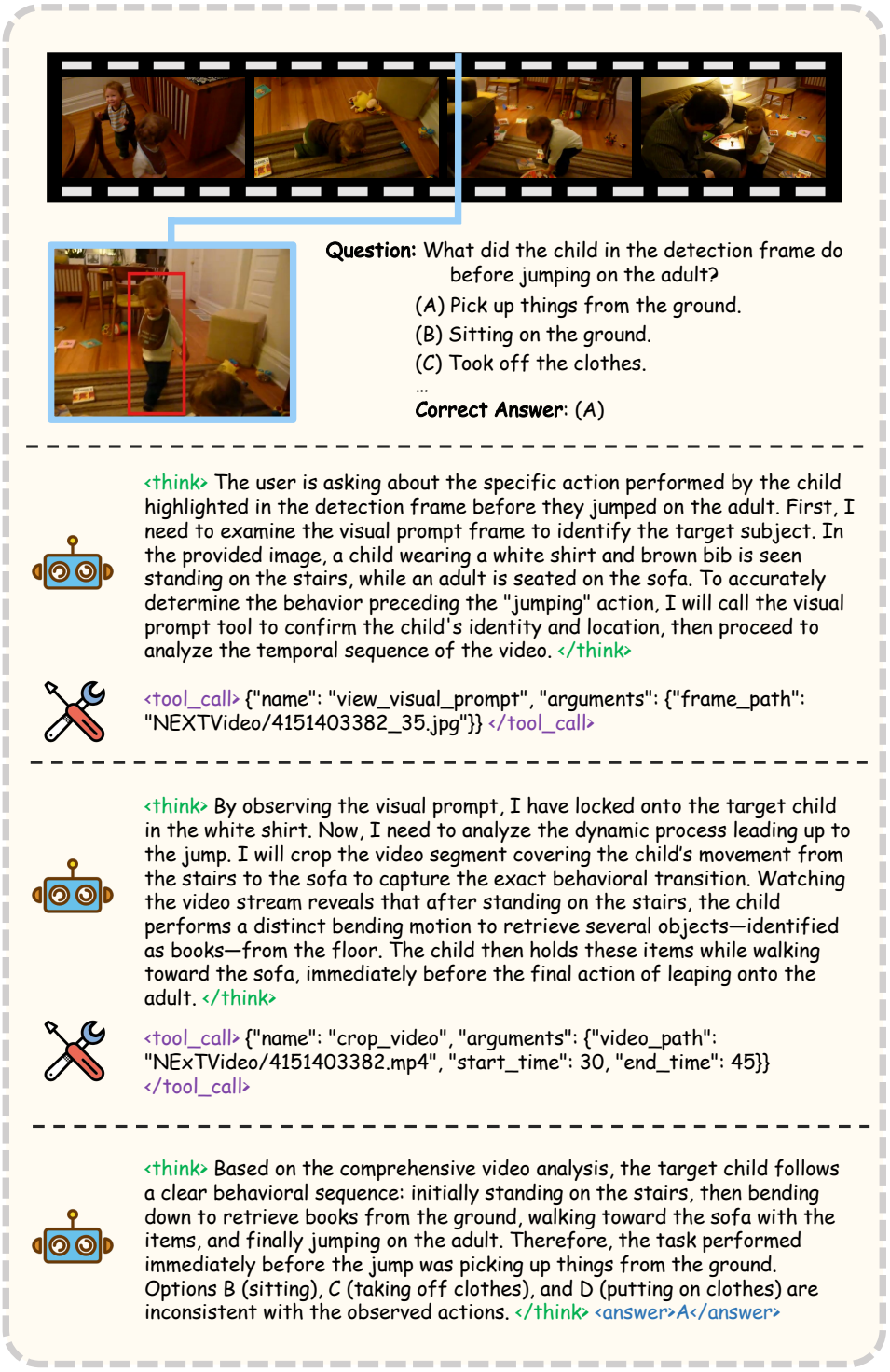

Figure 6: Case Study 1. The model invokes tools to proactively perceive instances and retrieve video segments, enabling instance-level video understanding tasks.





Question: What color is the top of the woman pointed by the red arrow?

(A) Green.
(B) Red.
(C) Blue.
...
Correct Answer: (A)

<think> The question requires identifying the color of a specific clothing item worn by a person marked with a red arrow. To provide an accurate answer, I first need to examine the visual prompt frame to locate the red arrow and identify the target individual among the subjects in the scene. By invoking the visual prompt tool, I can clearly observe the spatial layout and the specific person being pointed out, which is essential for determining the correct visual attribute. </think>

<tool_call> {"name": "view_visual_prompt", "arguments": {"frame_path": "MLVU/en_tv_24/en_tv_24.mp4_20250105_132503.472.jpg"}} </tool_call>

<think> After examining the visual prompt, the red arrow is clearly pointing to a woman standing within the scene. Focusing on her attire as indicated, her top exhibits a distinct green hue. The lighting in the scene is sufficient to confirm that the color is vibrant and consistent with the "Green" category. Comparing this observation with the available options, it is evident that the attribute matches the first choice. </think> <answer>A</answer>

Figure 7: Case Study 2. The question only requires visual cue information, so the model adaptively invokes only the visual cue tool, avoiding unnecessary tool calls.

D.2 PROMPTS

In this section, we illustrate all the prompts used in our paper.

D.2.1 TEXT FILTERING PROMPT

This prompt performs rapid pre-screening of QA samples to remove questions unsuitable for visual prompting (e.g., camera movements, scene backgrounds, counting). See Figure 8.

D.2.2 VIDEO VERIFICATION PROMPT

This prompt guides five-step reasoning with the video: target extraction, uniqueness judgment, temporal localization, QA rewriting, and visual prompt type recommendation. See Figure 9.

D.2.3 RENDERING AND REWRITING PROMPT

This prompt replaces target descriptions with generic visual prompt references, ensuring questions cannot be answered without visual prompting. See Figure 10.

Text Filtering Prompt

System Prompt:

Task Definition:
You are a data filter for visual prompting QA construction. Given ONLY the text of a question-answer pair (no video), quickly determine whether the question is potentially suitable for a visual-prompted QA task — i.e., whether it targets a specific, concrete visual entity (person / object / animal) that could in principle be highlighted by a bounding box, arrow, or contour on a video frame.

Reject (`is_valid: false`) — clear disqualifiers
Reject if the question clearly falls into ANY of the following types:

- **Camera / cinematography**: asks about camera movement, angle, zoom, transition, or how the video is shot. *e.g., "How does the camera move?", "What is the initial camera view?"*
- **Scene / background / setting**: asks about the overall scene, background, environment, setting, or atmosphere — not a specific foreground entity. *e.g., "What can be seen in the background?", "What is the primary setting of the video?"*
- **Overall / generic activity**: asks what the video is mainly about, without targeting a specific entity. *e.g., "What is the main activity in the video?", "What is the main focus?"*
- **Counting**: asks how many of something, rather than targeting one specific entity. *e.g., "How many shelves are visible?", "How many children are there?"*
- **Abstract / non-visual**: asks about abstract concepts, emotions, reasons without referencing a concrete visual entity; or targets non-visual properties (sound, smell, text labels, brand names on packaging). *e.g., "What indicates the cooking process?", "Which brand is on the jar?"*
- **Lighting / color of environment**: asks about ambient lighting changes or the color of background elements (walls, countertops) rather than a specific object.

Accept (`is_valid: true`):
Accept if the question appears to target a specific, concrete visual entity:

- A person described by clothing, role, or action: *"the person in the gray hoodie"*, *"the worker with the helmet"*
- A specific named tool or object: *"the wooden spoon"*, *"the blue measuring cup"*, *"the needle"*
- A specific animal: *"the dog"*, *"the kitten"*, *"the calf"*
- A question asking WHERE / WHAT COLOR / WHAT ACTION / WHY about a clearly identifiable entity

When in doubt → accept (`is_valid: true`). The downstream video analysis performs stricter uniqueness checking.
Output ONLY valid JSON, no extra text.
Output Format:

Accept: `{"is_valid": true}`
Reject: `{"is_valid": false, "reason": "one short sentence"}`

User Prompt:

Judge whether the following QA pair targets a specific visual entity suitable for visual prompting (text-only, no video).
Here is the question: `{question}`
Here is the options: `{options}`
Here is the answer: `{answer}`

Output JSON only.

Figure 8: Prompt of Text Filtering.

Video Verification Prompt

System Prompt:

You are a video understanding expert specializing in visual prompting data construction.

Your task: Given a video and an existing question-answer pair about the video, analyze whether the question targets a specific, uniquely identifiable object/person in the video, and if so, produce structured metadata for constructing visual-prompted QA data.

Definitions:

- **Target**: The primary object or person that the question is asking about.

- **Uniqueness**: The target must be visually distinguishable from all other entities in the video. If multiple similar entities exist (e.g., "a climber" when there are several climbers), the target lacks uniqueness unless the description includes differentiating attributes (e.g., "the climber in the red jacket").

- **Visual Prompt**: A visual annotation (e.g., rectangle, contour, arrow, etc.) overlaid on a video frame to unambiguously indicate the target, replacing textual descriptions.

Workflow:

Step 1: Target Extraction & Uniqueness Judgment

Identify the target entity from the question. Determine:

- Is the question about a specific visual entity (person/object/animal)? Questions about abstract concepts, events, scenes, or the whole video are NOT target-specific.
- Is this target uniquely identifiable in the video? Watch the video and verify that no other entity could be confused with the target.

Output `is_valid: false` if either condition fails, with a brief reason.
If ambiguity exists, prefer rejecting (`is_valid=false`) over guessing.

Examples:

- *"What colour shirt is the girl playing the violin wearing?"* $\rightarrow$ valid if exactly one such girl is identifiable.

- A tag like *"climber scaling vertical rock face"* $\rightarrow$ invalid if multiple climbers appear without disambiguation.

Step 2: Generate Target Tag

Produce a concise English tag (3-10 words) that:

- Uniquely identifies the target among all entities in the video
- Describes stable visual attributes (clothing, color, size, position, species, etc.)
- Is suitable as a text prompt for a detection/segmentation model (e.g., SAM3)

Step 3: Temporal Localization

Identify the primary continuous time window where the target is visible and relevant to the question:

- If the target appears in multiple segments, select the single most relevant segment
- Timestamps in seconds, rounded to 1 decimal place

Step 4: QA Rewriting

Rewrite the question by replacing the target's textual description with a visual prompt `<vp>`:

- Original: *"What does the man playing the drums do with his feet?"*
- Rewritten: "`What does <vp> do with his feet?`"

For answer options: only rewrite options that explicitly mention the target; leave others unchanged.

Step 5: Visual Prompt Type Recommendation

Select the most appropriate visual prompt type for this target from:

- `rectangle`: Bounding box. Best for well-bounded, non-overlapping targets.
- `mask_contour`: Outline of segmentation mask. Best for irregularly shaped or partially occluded targets.
- `ellipse`: Elliptical highlight. Best for faces or compact targets.

- `triangle`: Triangular highlight around or pointing to the target.
- `scribble`: Freehand scribble over the target region.
- `point`: Single dot on target. Best for very small targets.
- `arrow`: Arrow pointing to target. Best for small or point-like targets.
- `set_of_mark`: Numeric label on target (e.g., "the person marked with 1"). Best when multiple targets need simultaneous disambiguation.

Choose based on target size, shape, and scene complexity.

Output Format:

Output ONLY valid JSON, no extra text.

Here is a success sample:

```
{
  "is_valid": true,
  "reason": null,
  "sample_id": "<copy from input if provided, else null>",
  "target_description": "the original textual description of the target",
  "tag": "concise unique English tag for detection/segmentation",
  "timestamp": {
    "start": 2.5,
    "end": 18.3
  },
  "rewritten_question": "question with target replaced by <vp>",
  "rewritten_options": ["A. option1", "B. option2", "..."],
  "rewritten_answer": "the correct answer (unchanged)",
  "visual_prompt_type": "rectangle"
}
```

When `is_valid` is false, only output:

```
{
  "is_valid": false,
  "reason": "explanation of why this QA is not suitable"
}
```

User Prompt:

Here is the video: `{video}`
Here is the question: `{question}`
Here is the options: `{options}`
Here is the answer: `{answer}`
Analyze this QA pair following your workflow. Determine if the question targets a uniquely identifiable entity in the video, and if so, produce the complete structured output.

Figure 9: Prompt of Video Verification.

Rendering and Rewrite Prompt

System Prompt:

You are an expert dataset writer for visual prompted video QA. Your job is to rewrite QA text so that answering requires the visual prompt on the frame, not the target's original textual attributes.

Context:

- Upstream data already contains: `question`, `options`

- The placeholder `<vp>` marks where the target reference should be replaced by a visual prompt description.

Core objective:

- Replace <vp> with a natural visual prompt phrase.
- Keep semantic meaning unchanged.

Hard constraints:

- Do NOT change factual intent of the question.
- Do NOT alter option semantics, order, labels, or correctness.
- If an option does not contain <vp>, keep it exactly the same (except minimal grammar smoothing if absolutely necessary).
- Preserve language style and fluency.
- Never reveal concrete target identity in rewritten text.

Visual prompt phrase policy:

Use generic, prompt-driven references:

- `rectangle` → "the target in the highlighted box"
- `mask_contour` → "the target outlined by the contour"
- `ellipse` → "the target inside the ellipse"
- `triangle` → "the target indicated by the triangle marker"
- `scribble` → "the target under the scribble mark"
- `point` → "the target marked by the point"
- `arrow` → "the target indicated by the arrow"
- `set_of_mark` → "the target marked with number {n}" (if number unknown, use "the numbered target")

Options handling:

- Rewrite `rewritten_answer` only when it contains <vp>.
- Otherwise keep it unchanged.

Output format:

```
{
  "question_refined": [question],
}
```

User Prompt:

Please rewrite this sample for visual prompt dependency.

Here is the visual prompt type: {visual_prompt_type}
Here is the question: {question}
Here is the options: {options}
Please rewrite the question and options according to the system instructions.

Figure 10: Prompt of Rendering and Rewriting.