VIDEORESEARCHER: SELF-IMPROVING TOOL DESIGN FOR LONG-VIDEO UNDERSTANDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Video agents have made substantial progress in long-video understanding. Yet effective video-agent systems require costly, time-consuming manual design and trial and error. Current self-improvement methods either refine low-impact prompts, recombine predefined micro-tools, or waste most harness-optimization effort on low-impact components. To bridge this gap, we target high-impact video-tool updates with **VideoResearcher**, a training-free multi-agent framework that autonomously designs, tests, and refines tools for video understanding, like a human researcher. VideoResearcher operates through dual Solving and Evolving Loops: it analyzes tool-use trajectories to identify capability gaps, coordinates specialized agents to develop and validate executable tools, and reuses evolved tools in subsequent video reasoning. Through iterative tool refinement and validation, it progressively strengthens evidence acquisition without updating model parameters. VideoResearcher achieves state-of-the-art performance among self-improving agents and is competitive with human-engineered agents, demonstrating a training-free paradigm for long-video understanding that expands agent capabilities through autonomous tool development while reducing costly manual engineering.

1 INTRODUCTION

Video understanding connects visual perception with language reasoning, supporting applications from interactive assistants to embodied systems. Advances in large multimodal models have expanded these capabilities, enabling increasingly effective video dialogue and question answering (Li et al., 2023; Zohar et al., 2025). However, long videos introduce a harder requirement: models must locate relevant events and connect evidence scattered across time (Wu et al., 2024). Video agents largely address this challenge through iterative planning, tool use, and evidence gathering, allowing them to decide what to inspect next rather than answer from a single set of observations. Their results show effective agentic reasoning for long-video understanding (Wang et al., 2024c; Lin et al., 2026).

Existing video agents improve evidence acquisition through purpose-built tools. VideoAgent (Wang et al., 2024c) uses vision-language models (VLMs) and retrieval tools to iteratively gather question-relevant information; DVD (Zhang et al., 2025c) autonomously selects search tools over a multi-granular video database; and VideoSeek (Lin et al., 2026) progressively localizes critical evidence through overview, skim, and focus tools. These approaches demonstrate the value of tool-assisted reasoning, but developing effective toolboxes requires researchers to identify needed operations, diagnose tool limitations, and validate improvements to video reasoning. As illustrated on the left of Fig. 1, researchers repeatedly execute the system, inspect failure trajectories, and revise tools, manually translating execution feedback into implementation changes. Automating this process could reduce manual development effort and enable toolboxes to improve through use.

A growing body of work seeks to automate this development process through agent self-improvement. In general-purpose tasks, SkillOpt (Yang et al., 2026) refines tool-use guidance, Meta-Harness (Lee et al., 2026) optimizes whole agent-system code, and AutoResearch (Karpathy, 2026) automates code experimentation for model training. For long-video understanding, META (Huang et al., 2026) composes successful execution patterns into reusable macro-tools and extracts usage constraints from failures. These efforts demonstrate the potential to automate parts of the development cycle, but their updates may offer limited gains in long-video understanding. Better tool-use instructions (Yang et al., 2026) or new combinations of predefined tools (Huang et al., 2026) cannot directly address

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.

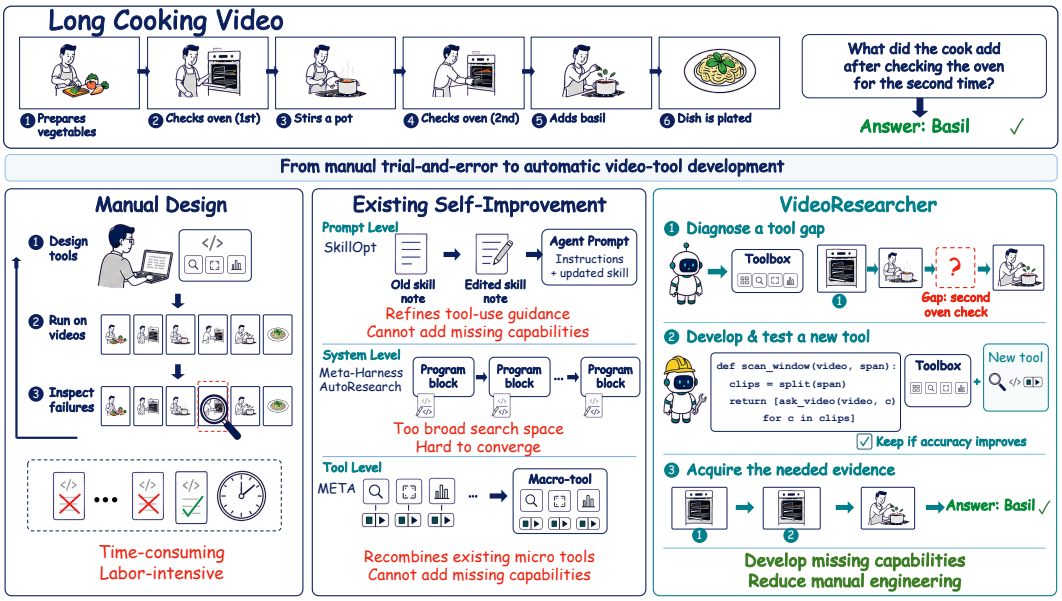

Figure 1: Motivation of VideoResearcher. Video-tool development remains labor-intensive. Existing self-improvement falls short: prompts have limited impact, system-wide search wastes most effort, and recombination remains bounded by predefined capabilities. VideoResearcher targets high-impact video-tool updates while avoiding costly system-wide optimization.

deficiencies in the underlying tools. Changing the entire system (Lee et al., 2026; Karpathy, 2026) allows broader exploration, but may largely waste effort on unrelated components. This raises a question: how can we enable more efficient self-improvement for long-video understanding?

To address this question, we introduce **VideoResearcher**, a training-free multi-agent framework for targeted video-tool development. *Our key insight is that tool-level self-improvement addresses evidence-acquisition deficiencies more efficiently than instruction refinement or tool recombination, avoiding system-level redesign.* Our framework consists of two coupled loops: a Solving Loop that gathers execution experience and an Evolving Loop that uses this experience to improve video tools. The Solving Loop separates reasoning, tool execution, and video perception, allowing the agent to actively gather evidence through VLM-based video tools. The Worker and perception model remain fixed, making the intermediate toolbox the sole target of self-improvement. Within this scope, the Evolving Loop coordinates specialized agents to develop and validate tool changes based on observed failures. The two loops connect problem solving with tool development, enabling targeted improvements without modifying the surrounding agent architecture or model parameters.

Extensive experiments on LVBench, LongVideoBench, and Video-MME evaluate the effectiveness of VideoResearcher. VideoResearcher outperforms the evaluated same-base self-improvement methods in overall accuracy on all three benchmarks, raising the Base agent's average accuracy from 72.1% to 74.5%. These results demonstrate effective targeted tool development for long-video understanding.

Our contributions are threefold:

- We introduce VideoResearcher, a training-free multi-agent framework that automates video-tool development for long-video understanding, reducing reliance on manual engineering.
- We propose failure-driven tool development, where specialized agents translate evidence gaps into tool specifications and executable code, retaining updates that improve answer accuracy.
- We demonstrate state-of-the-art performance among evaluated self-improving agents on three long-video benchmarks and show how autonomous tool development strengthens evidence acquisition.

2 Related Work

2.1 Long-Video Understanding

Long-video understanding depends on representing extended videos and finding relevant evidence. Video-language models benefit from unified visual representations (Li et al., 2025; Wang et al., 2024b; Bai et al., 2025) and richer captioning and instruction data (Chen et al., 2024; Zhang et al., 2025d). Long-video methods extend this foundation through hour-scale training (Chen et al., 2025; Lin et al., 2025), context transfer (Zhang et al., 2025b; Wei & Chen, 2025), and efficient visual representations (Shen et al., 2025; Shu et al., 2025; Xu et al., 2025; Ye et al., 2025; Wang et al., 2025b). Pooling and token compression reduce redundant inputs (Xu et al., 2024; Tao et al., 2025; Yang et al., 2025a), while reinforcement learning and self-distillation improve temporal reasoning and evidence use (Feng et al., 2025; Wang et al., 2026a). Visual Contrastive Self-Distillation derives on-policy training targets by contrasting predictions conditioned on original and content-erased images (Liang et al., 2026). HourVideo and Video-Holmes show that longer context alone is insufficient: models must also recall events and connect scattered clues (Chandrasegaran et al., 2024; Cheng et al., 2025).

Retrieval and memory make this evidence easier to access. Methods organize video into searchable documents, temporal trees, or dynamic memories (Ma et al., 2025; Wang et al., 2025c; Pang & Wang, 2025; Luo et al., 2025; Zuo et al., 2025); others retain past observations (He et al., 2024; Fan et al., 2024), retrieve from video collections (Jeong et al., 2025), or query hierarchical graph memories (Chen et al., 2026b). Agents further control evidence acquisition during reasoning: VideoAgent (Wang et al., 2024c) retrieves observations, DVD (Zhang et al., 2025c) searches a multi-granular database, and VideoSeek (Lin et al., 2026) moves from overview to focused inspection. Related agents adapt temporal search (Yang et al., 2025b; Yin et al., 2026; Tian et al., 2025), observation granularity (Wang et al., 2026b; Li et al., 2026a), or multi-agent coordination (Xu et al., 2026; Yan et al., 2026; Chen et al., 2026a). These methods improve evidence access, but their available operations are generally specified manually before execution. VideoResearcher automates this cycle: diagnosing missing capabilities, implementing tool changes, and testing them against the current toolbox.

2.2 Agent Evolution

Agent evolution improves guidance, system code, or reusable capabilities through execution feedback. Self-Refine revises outputs through self-feedback (Madaan et al., 2023); Reflexion retains verbal lessons across attempts (Shinn et al., 2023). DSPy, TextGrad, and GEPA optimize language-model programs through task evaluations or textual feedback (Khattab et al., 2024; Yuksekgonul et al., 2025; Agrawal et al., 2026). SkillOpt refines reusable tool-use instructions (Yang et al., 2026); Trace2Skill extracts trajectory lessons (Ni et al., 2026), while EvoSkill and CoEvoSkills develop skills through failure analysis or co-evolving verification (Alzubi et al., 2026; Zhang et al., 2026a). Skills may contain instructions, code, or both, requiring evaluation in use (Jiang et al., 2026; Li et al., 2026b).

Broader approaches optimize the agent system itself. Meta-Harness and VeRO improve the code surrounding model calls (Lee et al., 2026; Ursekar et al., 2026); SkillOpt-Lite and EvoTest also refine agent behavior from trajectories (Shen et al., 2026b; He et al., 2026). Architecture and workflow search modify roles, coordination, and execution structure (Hu et al., 2025; Zhang et al., 2025a; Song et al., 2026; Zhang et al., 2026b). Related program-evolution systems evaluate code changes in mathematical, algorithmic, or model-training tasks (Romera-Paredes et al., 2024; Novikov et al., 2025; Sharma, 2025; Zelikman et al., 2024; Karpathy, 2026). These methods offer broad flexibility; VideoResearcher instead focuses on executable tools that acquire video evidence.

Executable tool learning is the closest line of work. LATM and CREATOR separate tool creation from use (Cai et al., 2024; Qian et al., 2023); Voyager accumulates code skills from environment feedback (Wang et al., 2024a), and AgentOptimizer improves callable functions while freezing the language model (Zhang et al., 2024). ToolMaker and CASCADE construct tools or scientific skills from external resources (Wölflein et al., 2025; Huang et al., 2025), while experience reuse can develop tools for multimodal reasoning (Shen et al., 2026a). For long videos, META consolidates recurring micro-tool trajectories into macro-tools and derives usage priors from failures (Huang et al., 2026). VideoResearcher targets failure-directed video-tool development: structured diagnosis identifies an evidence gap, specialized agents implement a candidate, and development accuracy governs acceptance with fixed Worker and perception models, without parameter updates.

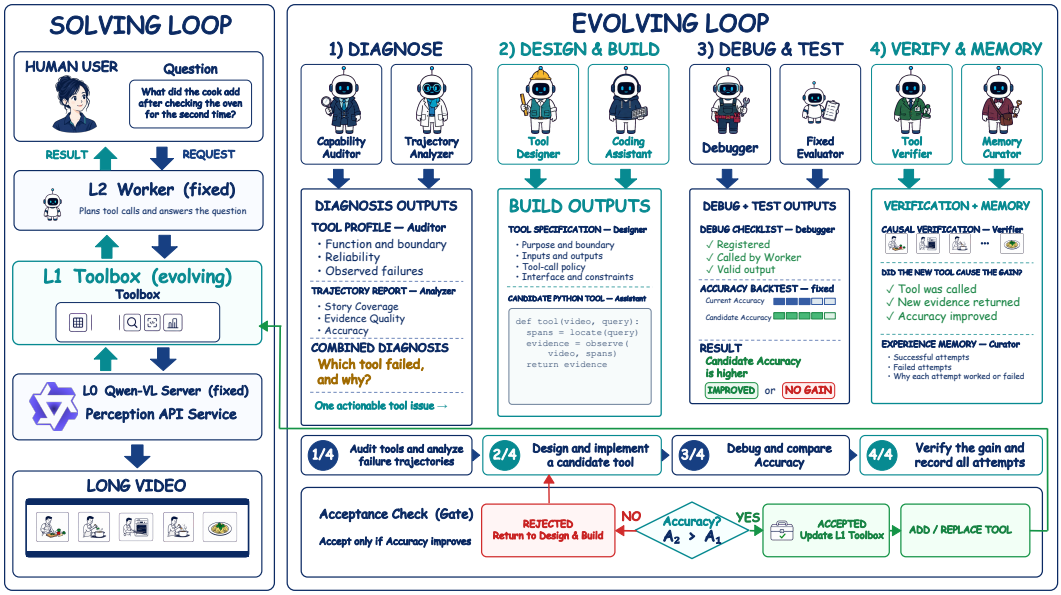

Figure 2: Overview of VideoResearcher. The Solving Loop connects a fixed Worker to a fixed perception model through an evolving toolbox. The Evolving Loop diagnoses trajectory failures, develops and tests executable tools, inspects their effects, and updates the toolbox.

3 METHOD

3.1 OVERVIEW

VideoResearcher couples Solving and Evolving Loops to improve video tools for long-video understanding (Fig. 2). The Solving Loop answers questions and records trajectories with the current toolbox; the Evolving Loop diagnoses failures, develops and tests executable tools, and retains updates that improve development accuracy. The Worker and perception model stay fixed; evolution changes how tools acquire evidence, without parameter updates or system redesign.

For a task $x_i = (V_i, q_i)$ with video V_i, question q_i, and reference answer y_i, the system at evolution round r is

$$\Pi^{(r)} = \big(\pi_{\mathrm{W}}, \mathcal{T}^{(r)}, P\big), \qquad \mathcal{T}^{(0)} = \{\texttt{ask_video}\}, \tag{1}$$

where π_{W} is the Worker, P is the perception model, and $\mathcal{T}^{(r)}$ is the toolbox, initially containing only the general-purpose `ask_video` tool. Fixing π_{W} and P isolates toolbox changes across rounds.

On a fixed development set $\mathcal{D}_{\mathrm{dev}}$, the loops collect experience and update the toolbox:

$$\begin{aligned} (\hat{y}_i^{(r)}, \tau_i^{(r)}) &= \mathrm{Solve}(x_i; \Pi^{(r)}), \\ \mathcal{E}^{(r)} &= \{(x_i, y_i, \hat{y}_i^{(r)}, \tau_i^{(r)})\}_{(x_i,y_i)\in\mathcal{D}_{\mathrm{dev}}}, \\ \big(\mathcal{T}^{(r+1)}, \mathcal{M}^{(r+1)}\big) &= \mathrm{Evolve}\Big(\mathcal{T}^{(r)}, \mathcal{M}^{(r)}, \mathcal{E}^{(r)}; A_{\mathrm{dev}}\Big), \end{aligned} \tag{2}$$

Experience $\mathcal{E}^{(r)}$ comprises tasks, reference answers, predictions $\hat{y}_i^{(r)}$, and trajectories $\tau_i^{(r)}$; memory $\mathcal{M}^{(r)}$, initially empty, records previous development attempts. The Evolving Loop develops candidates from this experience, accepting only those that improve development-set accuracy A_{dev}. Sections 3.2 and 3.3 detail trajectory collection and multi-agent tool development. Across rounds, evolution updates only $\mathcal{T}^{(r)}$ and $\mathcal{M}^{(r)}$, leaving π_{W} and P unchanged.

3.2 SOLVING: HIERARCHICAL TOOL USE

In the Solving Loop (Fig. 2), the Worker (L2) requests evidence, tools (L1) acquire it, and the perception model (L0) inspects video. It interleaves reasoning and action (Yao et al., 2023) with active evidence seeking (Wang et al., 2026b), evolving observation procedures with both models fixed.

Worker-guided reasoning. The Worker accesses video only through tools. At step j, it uses the question and interaction history $h_{i,<j}^{(r)}$ to select a tool $g_{ij}^{(r)} \in \mathcal{T}^{(r)}$ and arguments $a_{ij}^{(r)}$:

$$(g_{ij}^{(r)}, a_{ij}^{(r)}) = \pi_{\mathrm{W}}\left(q_i, h_{i,<j}^{(r)}; \mathcal{T}^{(r)}\right). \tag{3}$$

The Worker appends each observation to its history, then requests more evidence or produces $\hat{y}_i^{(r)}$. The Solving Loop caps this interaction at ten tool-use turns per question (Sec. 4.1).

Tool-mediated perception. Tools translate evidence requests into perception calls. The initial tool exposes the perception model directly:

$$\texttt{ask_video}(V_i; p, s, e, \rho) = P(V_i, p, s, e, \rho). \tag{4}$$

Here, p is the visual query passed as `question`, s and e are optional start and end times in seconds, and ρ is the sampling rate in frames per second (FPS). The tool defaults to the full video at 2 FPS and returns a textual observation. Evolved tools use the same perception interface but may define their own arguments, combine targeted calls, and aggregate observations into reusable procedures. Each evolved tool operates under a fixed budget of at most five perception calls and 512 sampled frames (Sec. 4.1), which bounds its inference cost per invocation.

Trajectory collection. Each trajectory preserves the interactions that produced an answer. For tool observation $o_{ij}^{(r)}$, we record

$$\begin{aligned} o_{ij}^{(r)} &= g_{ij}^{(r)}\left(V_i, a_{ij}^{(r)}; P\right), \qquad c_{ij}^{(r)} = \left(g_{ij}^{(r)}, a_{ij}^{(r)}, o_{ij}^{(r)}\right), \\ \tau_i^{(r)} &= \left(c_{i1}^{(r)}, \ldots, c_{iM_i}^{(r)}, \hat{y}_i^{(r)}\right), \end{aligned} \tag{5}$$

where M_i is the number of tool calls. The record includes underlying perception requests, giving the Evolving Loop the tool choices, arguments, and evidence needed for failure diagnosis. Failed trajectories thus reveal which evidence the current tools could not acquire.

3.3 Evolving: Multi-Agent Tool Development

The Evolving Loop assigns diagnosis, implementation, testing, and verification to specialized agents linked by structured reports (Fig. 2):

$$\mathcal{E}^{(r)} \xrightarrow[\mathcal{T}^{(r)}]{\text{diagnose}} d^{(r)} \xrightarrow[\mathcal{T}^{(r)},\,\mathcal{M}^{(r)}]{\text{design and build}} \widetilde{g}^{(r)} \xrightarrow{\text{debug and test}} b^{(r)} \xrightarrow[\mathcal{M}^{(r)}]{\text{verify and remember}} \left(\mathcal{T}^{(r+1)}, \mathcal{M}^{(r+1)}\right), \tag{6}$$

Here, $d^{(r)}$ is a diagnosis, $\widetilde{g}^{(r)}$ a candidate tool, $b^{(r)}$ a comparison against the current toolbox, and $\mathcal{M}^{(r)}$ the accumulated development experience. The lower labels specify inputs: diagnosis inspects the current toolbox, development uses it as the implementation base, and stored experience guides design and coding. Each candidate is thus linked to an observed failure and a recorded outcome. All roles are implemented with the Codex SDK (Sec. 4.1).

1) Diagnose. Diagnosis narrows execution failures to one tool-development target. The Capability Auditor produces a *Tool Profile* describing each tool's function, boundaries, reliability, and limitations. The Trajectory Analyzer combines this profile with trajectories to produce a *Trajectory Report* of recurring failures and answer outcomes. Its diagnostic criteria are *Story Coverage*, whether observations cover relevant events, and *Evidence Quality*, whether they provide useful evidence for answering the question. These criteria guide diagnosis rather than serve as acceptance objectives. The reports specify $d^{(r)}$: the failure, its triggering conditions, and the missing capability. For example, one diagnosis attributes wrong answers to unreliable coarse event localization, which motivates the Round-2 landmark index for locating coarse video events (Table 3(a)).

2) Design and build. Development turns $d^{(r)}$ into a specification and executable code. Using the *Tool Profile* and memory $\mathcal{M}^{(r)}$, the Tool Designer specifies the tool's purpose, input–output schema, boundaries, perception-call policy, and difference from existing tools in a *Tool Specification*. The

Table 1: Accuracy (%) on three representative long-video benchmarks. * marks methods evolved from the same Base, VideoResearcher (w/o Evolution); parentheses show changes from it. Average is computed over the three Overall scores. The best and second-best results are bolded and underlined.

Method	Venue	LVBench	LongVideoBench		Video-MME		Average
		Overall	Overall	Long	Overall	Long	Overall
Human-engineered agentic video systems							
VideoAgent (Wang et al., 2024c)	ECCV'24	29.3	–	–	–	46.4	–
VideoTree (Wang et al., 2025c)	CVPR'25	28.8	–	–	60.6	54.2	–
VideoLucy (Zuo et al., 2025)	NeurIPS'25	–	58.8	–	72.5	66.8	–
DVD (Zhang et al., 2025c)	NeurIPS'25	**74.2**	71.6	68.6	–	67.3	–
VideoSeek (Lin et al., 2026)	CVPR'26	<u>68.4</u>	–	**73.5**	–	70.1	–
VideoARM (Yin et al., 2026)	CVPR'26	–	73.7	69.2	<u>80.1</u>	**75.3**	–
VideoResearcher (w/o Evolution)	Ours	66.9	73.2	70.6	76.3	70.6	72.1
Self-improving agentic video systems							
META (Huang et al., 2026)	CVPR'26	–	63.6	–	77.7	–	–
SkillOpt* (Yang et al., 2026)	arXiv'26	64.8 (−2.1)	66.5 (−6.7)	63.8 (−6.8)	78.1 (+1.8)	74.1 (+3.5)	69.8 (−2.3)
Meta-Harness* (Lee et al., 2026)	arXiv'26	67.1 (+0.2)	<u>74.9</u> (+1.7)	70.9 (+0.3)	78.4 (+2.1)	73.6 (+3.0)	<u>73.5</u> (+1.4)
AutoResearch* (Karpathy, 2026)	GitHub'26	66.8 (−0.1)	73.0 (−0.2)	<u>73.4</u> (+2.8)	75.2 (−1.1)	73.3 (+2.7)	71.7 (−0.4)
VideoResearcher	Ours	<u>68.4</u> (+1.5)	**75.0** (+1.8)	71.6 (+1.0)	**80.2** (+3.9)	<u>74.7</u> (+4.1)	**74.5** (+2.4)

Coding Assistant implements it as a *Candidate Python Tool* $\widetilde{g}^{(r)}$ in an isolated toolbox. Changes may affect perception prompts, temporal decomposition, inspected intervals, frame selection, and output aggregation. The tool can be registered, invoked, and evaluated by the fixed Worker. For instance, the Round-2 landmark index splits a video into five segments, describes each with one perception call, and returns the ordered descriptions as five scene landmarks (Table 3(b)).

3) Debug and test. Testing checks both execution and answer accuracy. The Debugger runs smoke tests and records a *Debug Checklist* covering registration, execution, Worker invocation, schema compliance, and output validity. Failures return to the Coding Assistant for a bounded repair loop, so only candidates that execute correctly reach the accuracy backtest. After these checks pass, a fixed Evaluator compares the candidate and current toolboxes on the same development examples, using the same Worker, perception model, inference settings, and random seed. This *Accuracy Backtest* $b^{(r)}$ uses

$$A_{\text{dev}}(\mathcal{T}) = \frac{1}{|\mathcal{D}_{\text{dev}}|} \sum_{(x_i, y_i) \in \mathcal{D}_{\text{dev}}} \mathbb{I}[\hat{y}_i(\mathcal{T}) = y_i]\,. \tag{7}$$

Diagnosis and selection use this development set; final benchmarks remain separate (Sec. 4).

4) Verify and remember. Verification examines how a candidate changed the evidence. The Tool Verifier's *Effect Verification Report* records candidate invocation, new observations, and their relation to corrected or regressed answers. This trajectory-based explanation of the backtest is not an independent causal experiment. It never overrides the sole acceptance criterion, an accuracy gain:

$$\mathcal{T}^{(r+1)} = \begin{cases} \widetilde{\mathcal{T}}^{(r+1)}, & A_{\text{dev}}(\widetilde{\mathcal{T}}^{(r+1)}) > A_{\text{dev}}(\mathcal{T}^{(r)}), \\ \mathcal{T}^{(r)}, & \text{otherwise}, \end{cases} \tag{8}$$

where $\widetilde{\mathcal{T}}^{(r+1)} = \text{Update}(\mathcal{T}^{(r)}, \widetilde{g}^{(r)})$ is the isolated candidate toolbox. The Memory Curator appends the attempt's status, design, result, and lesson to *Experience Memory*, producing $\mathcal{M}^{(r+1)}$. Accepted tools enter the next Solving Loop; rejected attempts inform future design and coding. Later rounds can thus build on successful designs and avoid repeating failed ones. After the last round, the toolbox is frozen and used unchanged for all benchmark evaluations.

4 Experiments

4.1 Experimental Setup

Benchmarks. We evaluate on three long-video benchmarks. LVBench (Wang et al., 2025a) comprises 103 videos (each over 30 minutes; mean 68 minutes) and 1,549 multiple-choice questions

Table 2: Analysis of framework components, model choices, and tool evolution. Panel (d) breaks down LVBench by question type. ER, EU, KIR, TG, Rea, and Sum denote entity recognition, event understanding, key-information retrieval, temporal grounding, reasoning, and summarization.

(a) Framework components

Worker	Evolution	Perception	
		Qwen3-VL-8B	Qwen3.5-4B
×	×	58.0	57.6
✓	×	63.6	66.9
✓	✓	**66.8**	**68.4**

(b) Overall accuracy across benchmarks

Method	Benchmark		
	LVBench	LongVideoBench	Video-MME
VideoResearcher (w/o Evolution)	66.9	73.2	76.3
VideoResearcher (Round 2)	**68.9**	**75.5**	80.0
VideoResearcher (Round 7)	68.4	75.0	**80.2**

(c) Worker and perception models

Worker	Perception	
	Qwen3-VL-8B	Qwen3.5-4B
o4-mini	45.0	45.6
GPT-5.3-Codex	44.3	50.5
GPT-5.6-SOL	**66.8**	**68.4**

(d) LVBench accuracy by question type

Method	Question type						Overall
	ER	EU	KIR	TG	Rea	Sum	
VideoResearcher (w/o Evolution)	66.2	65.2	**77.3**	67.7	61.2	60.3	66.9
VideoResearcher (Round 2)	66.2	**68.3**	72.9	71.4	**65.7**	**77.6**	**68.9**
VideoResearcher (Round 7)	**66.6**	67.4	74.6	**72.3**	64.7	69.0	68.4

across six categories. LongVideoBench (Wu et al., 2024) contains 3,763 videos (up to 1 hour) and 6,678 questions for long-context retrieval and reasoning; we use its validation set. Video-MME (Fu et al., 2025) comprises 900 videos (254 hours) and 2,700 questions across six domains and three duration groups. We report Overall accuracy on all three benchmarks and long-duration subset accuracy (Long) for LongVideoBench and Video-MME, using subtitles only for LongVideoBench.

Implementation details. Unless otherwise noted, GPT-5.6-SOL and Qwen3.5-4B in instruct mode serve as the fixed Worker and perception model, respectively; Evolving roles use the Codex SDK. The Solving Loop allows at most 10 turns of tool use per question. The initial `ask_video` tool (Sec. 3.2) uses one perception call and at most 2,048 sampled frames. The Evolving Loop uses up to 20 rounds and 300 fixed ALLVB (Tan et al., 2025) development questions; videos average nearly two hours. Each evolved tool may make at most five perception calls and sample at most 512 frames in total. Only updates that strictly improve development accuracy are retained, and the toolbox is frozen for final testing on the three benchmarks, which are excluded from tool design and selection. The Qwen perception service was deployed using eight NVIDIA B200 GPUs.

Baselines and controlled comparison. Table 1 compares human-engineered and self-improving video agents. SkillOpt (Yang et al., 2026), Meta-Harness (Lee et al., 2026), AutoResearch (Karpathy, 2026), and VideoResearcher start from *VideoResearcher (w/o Evolution)*, called the *Base*, with the same Worker, perception model, inference limits, initial `ask_video` tool, 300 development questions, and 20-round evolution budget. This comparison varies the self-improvement strategy from a common starting agent. Human-engineered agents and META (Huang et al., 2026) use published configurations and scores under the closest matching benchmark settings; they provide external reference points rather than matched controls with identical Worker and perception models.

4.2 Main Results

Comparison with self-improving agents. VideoResearcher leads the evaluated self-improving agents on all three Overall scores and their average (Table 1). VideoResearcher outperforms SkillOpt, Meta-Harness, and AutoResearch by 4.7, 1.0, and 2.8 points in average accuracy, respectively, with notable gains including 8.5 points over SkillOpt on LongVideoBench and 5.0 points over AutoResearch on Video-MME. Compared with META, it improves Overall accuracy by 11.4 points on LongVideoBench and 2.5 points on Video-MME. These results are consistent with the optimization scope: SkillOpt refines instructions and lowers Base accuracy by 2.3 points; the evaluated Meta-Harness and AutoResearch runs mainly revise prompts, yielding limited gains. VideoResearcher adds two executable tools for better evidence acquisition (Table 3): *Neutral Scene Landmark Index* indexes coarse video events, and *Candidate Window Atomic Scan* inspects short clips near candidate events. The final toolbox raises Base's average accuracy from 72.1 to 74.5 (+2.4 points).

Table 3: Tool-evolution outcomes and an accepted executable tool. Panel (b) shows the Round-2 tool's contract, prompt, and core implementation; `Qwen` denotes the fixed perception model. Accuracy is measured on 300 development questions from ALLVB (Tan et al., 2025).

(a) Representative evolution outcomes

Round	Evolved tool	Dev. acc. Result
2	**Neutral Scene Landmark Index** *Target:* Unreliable coarse event localization *Function:* Build a five-part landmark index from about 512 frames	68.7 → 75.0 Accept
3	**Premise-Free Action Anchor Index** *Target:* Incomplete global action coverage *Function:* Extract early, middle, and late action anchors from five segments	75.0 → 70.7 Reject
7	**Candidate Window Atomic Scan** *Target:* Weak evidence near a candidate event *Function:* Scan five 20-s clips at 5 FPS for timestamped atomic events	75.0 → 76.7 Accept
14	**Appearance–Object Signature Index** *Target:* Broken cross-scene entity links *Function:* Link appearance, object, and interaction signatures across scenes	76.7 → 69.0 Reject
17	**Single-Context Midpoint Evidence Reel** *Target:* Evidence fragmented across calls *Function:* Inspect one midpoint window in a single context	76.7 → 71.7 Reject

(b) Accepted executable tool (Round 2)

Tool contract

Input: video path. Output: five ordered scene landmarks. Budget: five Qwen calls and approximately 512 sampled frames.

Evolved function

```
def neutral_scene_landmark_index(video):
    prompt = (
      "Analyze the clip and summarize "
      "visible events in temporal order. "
      "Infer roles and relationships "
      "from visible evidence. "
      "Consider alternatives and "
      "uncertainty before summarizing."
    )
    T = video_duration(video)
    dt, fps, segments = T / 5, 512 / T, []
    for i in range(5):
        start, end = (
          i * dt, min((i + 1) * dt, T))
        desc = Qwen(
          video, start, end, fps, prompt)
        segments.append({
          "segment": i + 1, "start": start,
          "end": end, "description": desc})
    return {"segments": segments}
```

Comparison with human-engineered agents. Against these published results, VideoResearcher achieves the highest Overall accuracy on LongVideoBench and Video-MME and ties VideoSeek for second on LVBench, behind DVD (Table 1). Its autonomously developed tools combine coarse indexing with focused local inspection, echoing DVD (Zhang et al., 2025c) and VideoSeek (Lin et al., 2026). These parallels suggest that tool evolution recovers effective strategies from manual design, although the published configurations do not isolate the effect of tool evolution.

4.3 Ablation Studies

Framework components. Agentic tool use and tool evolution both improve accuracy (Table 2(a)). With Qwen3-VL-8B, adding the Worker raises LVBench accuracy from 58.0 to 63.6, and tool evolution raises it to 66.8, an 8.8-point gain over direct perception. The complete framework reaches 68.4 with Qwen3.5-4B. Both perception models benefit from combining these components.

Worker and perception models. Stronger Workers substantially improve accuracy, consistent with prior findings (Zhang et al., 2025c; Chen et al., 2026b). Table 2(c) tests three Workers (o4-mini, GPT-5.3-Codex, and GPT-5.6-SOL) with two perception models using the same evolved toolbox, without further evolution. Weaker Workers even fall below direct perception (Table 2(a)), so a strong Worker is essential. Changing the perception model yields smaller gains: with GPT-5.6-SOL, switching from Qwen3-VL-8B to Qwen3.5-4B raises accuracy from 66.8 to 68.4.

Effects of evolved tools. Adding more tools does not necessarily yield further gains. Table 2(b) shows gains over Base on LVBench, LongVideoBench, and Video-MME of 2.0, 2.3, and 3.7 points at Round 2 (landmark index only), versus 1.5, 1.8, and 3.9 points at Round 7 after adding the atomic scan. Table 2(d) supports the tools' intended function: the Round-2 landmark index raises accuracy in locating when events occur (temporal grounding, TG) from 67.7 to 71.4 (+3.7 points).

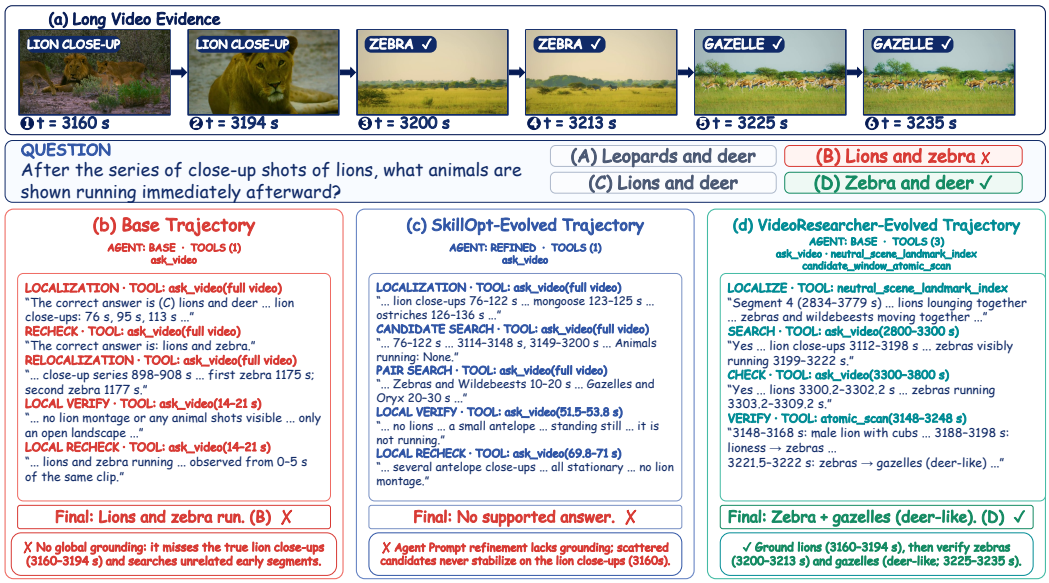

Figure 3: Tool-use trajectories for the same video and question. (a) Shared long-video evidence. (b) Base uses the default `ask_video` tool. (c) SkillOpt revises the Worker prompt with the same tool interface. (d) VideoResearcher's evolved tools retrieve decisive evidence and answer correctly.

4.4 Qualitative Analysis of Tool Evolution

Evolution outcomes. VideoResearcher proposes distinct tool behaviors for different evidence failures. Table 3(a) covers coarse indexing, action anchoring, local event scanning, cross-scene association, and context consolidation. The landmark index and atomic scan are accepted, raising development accuracy from 68.7 to 75.0 and then 76.7; the other three proposals reduce accuracy and are rejected. Panel (b) shows the Round-2 tool's contract and executable implementation. Thus, accepted tools expand the toolbox, while rejected proposals preserve unresolved problems and unsuccessful implementations for later evolution or manual design.

Trajectory case study. Figure 3 illustrates how a new tool changes acquired evidence and the final answer. Panels (b)–(d) compare Base, SkillOpt, and VideoResearcher. Base misses the decisive transition with `ask_video`; SkillOpt's revised Worker prompt retains the tool interface and still fails to retrieve sufficient evidence. The evolved indexing and local-scanning tools locate the relevant interval, recover the zebra-and-gazelle evidence, and enable VideoResearcher to answer correctly. This case connects an identified failure to a new capability, better evidence, and a corrected answer.

5 Conclusion

In this work, we introduced VideoResearcher, a training-free multi-agent framework that automates tool development for long-video understanding. By diagnosing evidence gaps in execution trajectories, developing executable tools, and validating accuracy gains, VideoResearcher turns observed failures into tested toolbox updates. This process strengthens evidence acquisition while keeping the Worker and perception model fixed, allowing agents to expand their capabilities without repeated manual tool design. Experiments on three benchmarks demonstrate state-of-the-art performance among self-improving agents, remaining competitive with human-engineered agents and raising average accuracy from 72.1% to 74.5%. These results highlight autonomous tool development as a practical route to more capable long-video agents. However, the current framework relies on textual trajectory assessment and repeated selection on one development set, and its evolution remains confined to tools. Future work will investigate independent candidate-selection data and extend evolution to agent architectures, enabling adaptation in both evidence acquisition and reasoning organization.

AI USE STATEMENT

In this work, generative AI tools were used solely to improve the readability and linguistic clarity of the manuscript. They were not used to formulate research ideas, design methods or experiments, generate or analyze data, implement the proposed methods, or interpret results. All substantive research contributions were made by the authors, who reviewed and verified all AI-assisted edits and take full responsibility for the final manuscript.

REFERENCES

Lakshya A Agrawal, Shangyin Tan, Dilara Soylu, Noah Ziems, Rishi Khare, Krista Opsahl-Ong, Arnav Singhvi, Herumb Shandilya, Michael J Ryan, Meng Jiang, et al. GEPA: Reflective prompt evolution can outperform reinforcement learning. In *ICLR*, 2026.

Salaheddin Alzubi, Noah Provenzano, Jaydon Bingham, Weiyuan Chen, and Tu Vu. EvoSkill: Automated skill discovery for multi-agent systems. *arXiv preprint arXiv:2603.02766*, 2026.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Tianle Cai, Xuezhi Wang, Tengyu Ma, Xinyun Chen, and Denny Zhou. Large language models as tool makers. In *ICLR*, 2024.

Keshigeyan Chandrasegaran, Agrim Gupta, Lea M Hadzic, Taran Kota, Jimming He, Cristóbal Eyzaguirre, Zane Durante, Manling Li, Jiajun Wu, and Li Fei-Fei. HourVideo: 1-hour video-language understanding. In *NeurIPS*, 2024.

Boyu Chen, Zikang Wang, Zhengrong Yue, Kainan Yan, Chenyun Yu, Yi Huang, Zijun Liu, Yafei Wen, Xiaoxin Chen, Yang Liu, Peng Li, and Yali Wang. VideoChat-M1: Collaborative policy planning for video understanding via multi-agent reinforcement learning. In *CVPR*, 2026a.

Cong Chen, Guo Gan, Kaixiang Ji, ZhaoYang Zhang, Zhen Yang, Guangming Yao, Hao Chen, Jingdong Chen, Yi Yuan, and Chunhua Shen. MemDreamer: Decoupling perception and reasoning for long video understanding via hierarchical graph memory and agentic retrieval mechanism. *arXiv preprint arXiv:2606.07512*, 2026b.

Lin Chen, Xilin Wei, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Bin Lin, Zhenyu Tang, et al. ShareGPT4Video: Improving video understanding and generation with better captions. In *NeurIPS*, 2024.

Yukang Chen, Fuzhao Xue, 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. In *ICLR*, 2025.

Junhao Cheng, Yuying Ge, Teng Wang, Yixiao Ge, Jing Liao, and Ying Shan. Video-Holmes: Can MLLM think like Holmes for complex video reasoning? *arXiv preprint arXiv:2505.21374*, 2025.

Yue Fan, Xiaojian Ma, Rujie Wu, Yuntao Du, Jiaqi Li, Zhi Gao, and Qing Li. VideoAgent: A memory-augmented multimodal agent for video understanding. In *ECCV*, 2024.

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. In *NeurIPS*, 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 *CVPR*, 2025.

Bo He, Hengduo Li, Young Kyun Jang, Menglin Jia, Xuefei Cao, Ashish Shah, Abhinav Shrivastava, and Ser-Nam Lim. MA-LMM: Memory-augmented large multimodal model for long-term video understanding. In *CVPR*, 2024.

Yufei He, Juncheng Liu, Yue Liu, Yibo Li, Tri Cao, Zhiyuan Hu, Xinxing Xu, and Bryan Hooi. EvoTest: Evolutionary test-time learning for self-improving agentic systems. In *ICLR*, 2026.

Shengran Hu, Cong Lu, and Jeff Clune. Automated design of agentic systems. In *ICLR*, 2025.

Jing Huang, Luyuan Chen, Zhijie Xu, Yadong Li, Xingzhong Xu, Siye Chen, Jie Liu, Ming Kong, and Qiang Zhu. META: Meta evolution of tool trajectory adaptation for long-video understanding. In *CVPR*, 2026.

Xu Huang, Junwu Chen, Yuxing Fei, Zhuohan Li, Philippe Schwaller, and Gerbrand Ceder. CASCADE: Cumulative agentic skill creation through autonomous development and evolution. *arXiv preprint arXiv:2512.23880*, 2025.

Soyeong Jeong, Kangsan Kim, Jinheon Baek, and Sung Ju Hwang. VideoRAG: Retrieval-augmented generation over video corpus. In *Findings of ACL*, 2025.

Yanna Jiang, Delong Li, Haiyu Deng, Baihe Ma, Xu Wang, Qin Wang, and Guangsheng Yu. SoK: Agentic skills–beyond tool use in LLM agents. *arXiv preprint arXiv:2602.20867*, 2026.

Andrej Karpathy. AutoResearch. GitHub repository: karpathy/autoresearch, 2026.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan A, Saiful Haq, Ashutosh Sharma, Thomas Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. DSPy: Compiling declarative language model calls into state-of-the-art pipelines. In *ICLR*, 2024.

Yoonho Lee, Roshen Nair, Qizheng Zhang, Kangwook Lee, Omar Khattab, and Chelsea Finn. Meta-Harness: End-to-end optimization of model harnesses. *arXiv preprint arXiv:2603.28052*, 2026.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-OneVision: Easy visual task transfer. *TMLR*, 2025.

Keliang Li, Yansong Li, Hongze Shen, Mengdi Liu, Hong Chang, and Shiguang Shan. LensWalk: Agentic video understanding by planning how you see in videos. In *CVPR*, 2026a.

KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. VideoChat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023.

Xiangyi Li, Yimin Liu, Wenbo Chen, Bingran You, Zonglin Di, Yifeng He, Shenghan Zheng, Kyoung Whan Choe, Jiankai Sun, Shuyi Wang, et al. SkillsBench: Benchmarking how well agent skills work across diverse tasks. *arXiv preprint arXiv:2602.12670*, 2026b.

Yijun Liang, Yunjie Tian, Yijiang Li, Yuqi Jia, Furong Huang, Tianyi Zhou, and Di Fu. Visual contrastive self-distillation. *arXiv preprint arXiv:2607.21556*, 2026.

Jingyang Lin, Jialian Wu, Ximeng Sun, Ze Wang, Jiang Liu, Yusheng Su, Xiaodong Yu, Hao Chen, Jiebo Luo, Zicheng Liu, et al. Unleashing hour-scale video training for long video-language understanding. In *NeurIPS*, 2025.

Jingyang Lin, Jialian Wu, Jiang Liu, Ximeng Sun, Ze Wang, Xiaodong Yu, Jiebo Luo, Zicheng Liu, and Emad Barsoum. VideoSeek: Long-horizon video agent with tool-guided seeking. In *CVPR*, 2026.

Yongdong Luo, Xiawu Zheng, Guilin Li, Shukang Yin, Haojia Lin, Chaoyou Fu, Jinfa Huang, Jiayi Ji, Fei Chao, Jiebo Luo, et al. Video-RAG: Visually-aligned retrieval-augmented long video comprehension. In *NeurIPS*, 2025.

Ziyu Ma, Chenhui Gou, Hengcan Shi, Bin Sun, Shutao Li, Hamid Rezatofighi, and Jianfei Cai. DrVideo: Document retrieval based long video understanding. In *CVPR*, 2025.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-Refine: Iterative refinement with self-feedback. In *NeurIPS*, 2023.

Jingwei Ni, Yihao Liu, Xinpeng Liu, Yutao Sun, Mengyu Zhou, Pengyu Cheng, Dexin Wang, Erchao Zhao, Xiaoxi Jiang, and Guanjun Jiang. Trace2Skill: Distill trajectory-local lessons into transferable agent skills. *arXiv preprint arXiv:2603.25158*, 2026.

Alexander Novikov, Ngân Vũ, Marvin Eisenberger, Emilien Dupont, Po-Sen Huang, Adam Zsolt Wagner, Sergey Shirobokov, Borislav Kozlovskii, Francisco JR Ruiz, Abbas Mehrabian, et al. AlphaEvolve: A coding agent for scientific and algorithmic discovery. *arXiv preprint arXiv:2506.13131*, 2025.

Ziqi Pang and Yu-Xiong Wang. MR. Video: MapReduce as an effective principle for long video understanding. In *NeurIPS*, 2025.

Cheng Qian, Chi Han, Yi Fung, Yujia Qin, Zhiyuan Liu, and Heng Ji. CREATOR: Tool creation for disentangling abstract and concrete reasoning of large language models. In *Findings of EMNLP*, 2023.

Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balog, M Pawan Kumar, Emilien Dupont, Francisco JR Ruiz, Jordan S Ellenberg, Pengming Wang, Omar Fawzi, et al. Mathematical discoveries from program search with large language models. *Nature*, 2024.

Asankhaya Sharma. OpenEvolve: an open-source evolutionary coding agent. GitHub repository: algorithmicsuperintelligence/openevolve, 2025.

Xiaoqian Shen, Yunyang Xiong, Changsheng Zhao, Lemeng Wu, Jun Chen, Chenchen Zhu, Zechun Liu, Fanyi Xiao, Balakrishnan Varadarajan, Florian Bordes, et al. LongVU: Spatiotemporal adaptive compression for long video-language understanding. In *ICML*, 2025.

Xintian Shen, Jiawei Chen, Lihao Zheng, Hao Ma, Tao Wei, and Kun Zhan. Evolving from tool user to creator via training-free experience reuse in multimodal reasoning. *arXiv preprint arXiv:2602.01983*, 2026a.

Yifei Shen, Bo Li, and Xinjie Zhang. SkillOpt-Lite: Better and faster agent self-evolution via one line of vibe. *arXiv preprint arXiv:2607.03451*, 2026b.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *NeurIPS*, 2023.

Yan Shu, Zheng Liu, Peitian Zhang, Minghao Qin, Junjie Zhou, Zhengyang Liang, Tiejun Huang, and Bo Zhao. Video-XL: Extra-long vision language model for hour-scale video understanding. In *CVPR*, 2025.

Weijia Song, Jiashu Yue, and Zhe Pang. ABSTRAL: Automatic design of multi-agent systems through iterative refinement and topology optimization. *arXiv preprint arXiv:2603.22791*, 2026.

Xichen Tan, Yuanjing Luo, Yunfan Ye, Fang Liu, and Zhiping Cai. ALLVB: All-in-one long video understanding benchmark. In *AAAI*, 2025.

Keda Tao, Can Qin, Haoxuan You, Yang Sui, and Huan Wang. DyCoke: Dynamic compression of tokens for fast video large language models. In *CVPR*, 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.

Varun Ursekar, Apaar Shanker, Veronica Chatrath, Yuan Xue, and Samuel Marc Denton. VeRO: A harness for agents to optimize agents. In *ICML*, 2026.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *TMLR*, 2024a.

Kaishen Wang, Dongdi Zhao, Yijun Liang, Dingqiang Ye, Ruibo Chen, Heng Huang, and Di Fu. Where to look matters: On-policy self-distillation for long-video understanding. *arXiv preprint arXiv:2608.25356*, 2026a.

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

Weihan Wang, Zehai He, Wenyi Hong, Yean Cheng, Xiaohan Zhang, Ji Qi, Ming Ding, Xiaotao Gu, Shiyu Huang, Bin Xu, et al. LVBench: An extreme long video understanding benchmark. In *ICCV*, 2025a.

Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. VideoAgent: Long-form video understanding with large language model as agent. In *ECCV*, 2024c.

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*, 2025b.

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 *CVPR*, 2025c.

Ziyang Wang, Honglu Zhou, Shijie Wang, Junnan Li, Caiming Xiong, Silvio Savarese, Mohit Bansal, Michael S. Ryoo, and Juan Carlos Niebles. Active video perception: Iterative evidence seeking for agentic long video understanding. In *CVPR Findings*, 2026b.

Hongchen Wei and Zhenzhong Chen. Visual context window extension: A new perspective for long video understanding. In *ACM MM*, 2025.

Georg Wölflein, Dyke Ferber, Daniel Truhn, Ognjen Arandjelović, and Jakob Nikolas Kather. LLM agents making agent tools. In *ACL*, 2025.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. LongVideoBench: A benchmark for long-context interleaved video-language understanding. In *NeurIPS*, 2024.

Lin Xu, Yilin Zhao, Daquan Zhou, Zhijie Lin, See Kiong Ng, and Jiashi Feng. PLLaVA: Parameter-free LLaVA extension from images to videos for video dense captioning. *arXiv preprint arXiv:2404.16994*, 2024.

Mingze Xu, Mingfei Gao, Shiyu Li, Jiasen Lu, Zhe Gan, Zhengfeng Lai, Meng Cao, Kai Kang, Yinfei Yang, and Afshin Dehghan. SlowFast-LLaVA-1.5: A family of token-efficient video large language models for long-form video understanding. In *COLM*, 2025.

Yichang Xu, Gaowen Liu, Ramana Rao Kompella, Tiansheng Huang, Sihao Hu, Fatih Ilhan, Selim Furkan Tekin, Zachary Yahn, and Ling Liu. A multi-agent perception-action alliance for efficient long video reasoning. In *CVPR*, 2026.

Haiyang Yan, Hongyun Zhou, Peng Xu, Xiaoxue Feng, and Mengyi Liu. Symphony: A cognitively-inspired multi-agent system for long-video understanding. In *CVPR*, 2026.

Chenyu Yang, Xuan Dong, Xizhou Zhu, Weijie Su, Jiahao Wang, Hao Tian, Zhe Chen, Wenhai Wang, Lewei Lu, and Jifeng Dai. PVC: Progressive visual token compression for unified image and video processing in large vision-language models. In *CVPR*, 2025a.

Yifan Yang, Ziyang Gong, Weiquan Huang, Qihao Yang, Ziwei Zhou, Zisu Huang, Yan Li, Xuemei Gao, Qi Dai, Bei Liu, Kai Qiu, Yuqing Yang, Dongdong Chen, Xue Yang, and Chong Luo. SkillOpt: Executive strategy for self-evolving agent skills. *arXiv preprint arXiv:2605.23904*, 2026.

Zeyuan Yang, Delin Chen, Xueyang Yu, Maohao Shen, and Chuang Gan. VCA: Video curious agent for long video understanding. In *ICCV*, 2025b.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *ICLR*, 2023.

Jiabo Ye, Haiyang Xu, Haowei Liu, Anwen Hu, Ming Yan, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. mPLUG-Owl3: Towards long image-sequence understanding in multi-modal large language models. In *ICLR*, 2025.

Yufei Yin, Qianke Meng, Minghao Chen, Jiajun Ding, Zhenwei Shao, and Zhou Yu. VideoARM: Agentic reasoning over hierarchical memory for long-form video understanding. In *CVPR*, 2026.

Mert Yuksekgonul, Federico Bianchi, Joseph Boen, Sheng Liu, Pan Lu, Zhi Huang, Carlos Guestrin, and James Zou. Optimizing generative AI by backpropagating language model feedback. *Nature*, 2025.

Eric Zelikman, Eliana Lorch, Lester Mackey, and Adam Tauman Kalai. Self-Taught Optimizer (STOP): Recursively self-improving code generation. In *COLM*, 2024.

Hanrong Zhang, Shicheng Fan, Henry Peng Zou, Yankai Chen, Zhenting Wang, Jiayu Zhou, Chengze Li, Wei-Chieh Huang, Yifei Yao, Kening Zheng, et al. CoEvoSkills: Self-evolving agent skills via co-evolutionary verification. *arXiv preprint arXiv:2604.01687*, 2026a.

Jenny Zhang, Shengran Hu, Cong Lu, Robert Lange, and Jeff Clune. Darwin Gödel machine: Open-ended evolution of self-improving agents. In *ICLR*, 2026b.

Jiayi Zhang, Jinyu Xiang, Zhaoyang Yu, Fengwei Teng, Xionghui Chen, Jiaqi Chen, Mingchen Zhuge, Xin Cheng, Sirui Hong, Jinlin Wang, Bingnan Zheng, Bang Liu, Yuyu Luo, and Chenglin Wu. AFlow: Automating agentic workflow generation. In *ICLR*, 2025a.

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. *TMLR*, 2025b.

Shaokun Zhang, Jieyu Zhang, Jiale Liu, Linxin Song, Chi Wang, Ranjay Krishna, and Qingyun Wu. Offline training of language model agents with functions as learnable weights. In *ICML*, 2024.

Xiaoyi Zhang, Zhaoyang Jia, Zongyu Guo, Jiahao Li, Bin Li, Houqiang Li, and Yan Lu. Deep video discovery: Agentic search with tool use for long-form video understanding. In *NeurIPS*, 2025c.

Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. LLaVA-Video: Video instruction tuning with synthetic data. *TMLR*, 2025d.

Orr Zohar, Xiaohan Wang, Yann Dubois, Nikhil Mehta, Tong Xiao, Philippe Hansen-Estruch, Licheng Yu, Xiaofang Wang, Felix Juefei-Xu, Ning Zhang, et al. Apollo: An exploration of video understanding in large multimodal models. In *CVPR*, 2025.

Jialong Zuo, Yongtai Deng, Lingdong Kong, Jingkang Yang, Rui Jin, Yiwei Zhang, Nong Sang, Liang Pan, Ziwei Liu, and Changxin Gao. VideoLucy: Deep memory backtracking for long video understanding. In *NeurIPS*, 2025.