# VideoMind: Thinking in Steps for Long Video Understanding

**Shubhang Bhatnagar**[1,2,*], **Renxiong Wang**[2], **Kapil Krishnakumar**[2], **Adel Ahmadyan**[2], **Zhaojiang Lin**[2], **Lambert Mathias**[2], **Xin Luna Dong**[2], **Babak Damavandi**[2], **Narendra Ahuja**[1], **Seungwhan Moon**[2]
[1]University of Illinois Urbana-Champaign, [2]Meta
[*]Work done as an intern at Meta.
**Correspondence:** sb56@illinois.edu

## Abstract

Multimodal Large Language Models (MLLMs) struggle with Long Video Understanding (LVU) due to their limited context window and the distributed nature of salient information across many redundant frames. To address this, we present VideoMind, a novel training free framework for LVU designed to mimic a human reasoning process. The framework is orchestrated by an MLLM that breaks down a user's query into a series of simpler, actionable sub-queries. For each sub query, the MLLM reconfigures itself by invoking specialized 'modes' that are instantiations of the same MLLM, but with appropriately tailored context for the given sub query to extract targeted evidence. After gathering this evidence, the model resumes its role as the orchestrator which evaluates the results and decides if an answer is complete or if it must refine its strategy by engaging further modes with new context. Our specialized operational modes include: 1) a Multi-Scale Temporal Search mode to identify and summarize relevant video sub-snippets at varying time scales, and 2) a Single-Frame Visual Detail mode for precise spatial localization of objects. This dynamic allocation of computation yields state-of-the-art results on the Video-MME, LongVideo, and MLVU benchmarks, achieving 77.6% performance on Video MME using Qwen 2.5 72B (4.8% enhancement) while also yielding a 5% improvement on Llama 4 Scout.

## 1 Introduction

Long Video Understanding (LVU) represents a critical frontier in computer vision, essential for applications requiring sustained attention over extended timelines. These applications range from complex activity recognition (Guo et al., 2022; Shao et al., 2020) and automated content summarization (Lee et al., 2025) to interactive archival search (Rossetto et al., 2025). Unlike short clips, typical long-form videos span several minutes to hours, containing vast amounts of redundant information interspersed with sparse, highly salient events. However, this sparsity of salient information poses a fundamental challenge to current Multimodal Large Language Models (MLLMs) (AI, 2025; Gemini et al., 2024; Liu et al., 2023; OpenAI, 2023; Bai et al., 2025), as they are unable to attend to such events in such long contexts (usually processed as uniformly sampled input frames), pushing them beyond architectural limits.

To address this challenge, we introduce VideoMind, a novel, training-free agentic framework that reframes the MLLM as a human-like reasoning engine that actively interrogates the video rather than passively consuming frames or captions at once. Given a complex user query, the controller MLLM first decomposes it into a sequence of focused, actionable sub-queries, mirroring how a human would break a problem into manageable steps. It then solves these sub-problems by invoking a set of specialized modes, which are instantiations of the same MLLM tailored with minimal, relevant video context (e.g., a few frames) for that specific sub-task. The MLLM in these modes provides concise textual answers for the sub queries. The MLLM then resumes its role as the orchestrator and reasons over the text, either synthesizing a final answer or iteratively refining its strategy by engaging into another mode with appropriately modified sub-queries and video contexts. This represents a form of self-specialization, where the model's generalist abilities are harnessed to create expert functions.

Specifically, our framework operationalizes this approach with two such 'modes' to interact with the video: (1) The Multi-Scale Temporal Search mode, which is designed to efficiently identify video segments relevant to a given sub query at a time scale. (2) The Spatial Detail mode which extracts fine-grained visual evidence from specific frames. Both these modes enable a coarse-to-fine workflow to help focus the MLLM on the most salient temporal



*Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 5: Industry Track)*, pages 406–416
March 25-27, 2026 

and spatial regions.

We demonstrate the effectiveness of VideoMind through extensive experiments on three diverse LVU benchmarks: Video-MME (Fu et al., 2025), LongVideoBench (Wu et al., 2024), and MLVU (Cui et al., 2024). VideoMind consistently elevates powerful base MLLMs, boosting Qwen 2.5 72B (Bai et al., 2025) by 4.8% to 77.6% and Llama 4 Scout (AI, 2025) by 5.0% to 67.8% in accuracy on Video-MME, with gains concentrated in complex multi-step temporal reasoning tasks (e.g., Action Reasoning, Temporal Perception). Comprehensive ablation studies further validate our hierarchical MLLM mode design.

Our primary contributions are:

- **VideoMind**, a novel, training-free agentic framework for long video understanding that empowers a base MLLM to dynamically decompose complex queries and iteratively seek evidence by shifting into specialized reasoning modes, and reason over the gathered textual evidence.
- A set of operational modes that allows the base MLLM to dynamically re-purpose its own capabilities to interact with the salient parts of the video for (1) a **Multi-Scale Temporal Search** and (2) a **Spatial Detail Analysis**, in a coarse-to-fine LVU workflow.
- **State-of-the-art performance on LVU benchmarks** demonstrating the effectiveness of our approach, with VideoMind achieving 77.6% in accuracy on the Video-MME benchmark using the Qwen 2.5 VL 72B backbone.

## 2 Related Work

Our work, VideoMind, builds upon advancements in Multimodal Large Language Models (MLLMs), Long Video Understanding (LVU), and the emerging paradigm of agentic AI systems. End-to-end MLLMs that process video as a uniformly sampled input frames (Zhang et al., 2024; Li et al., 2024a) become prohibitively token-intensive for hour-long content, struggling to scale effectively due to quadratic attention complexity (Liu et al., 2024a), while choosing too few frames risks leaving out salient events. To mitigate this, Ma et al. (2024); Shen et al. (2024) propose strategies based on token compression, but these lossy approaches cannot guarantee the selection of question-relevant tokens. Another promising direction has emerged in the form of video agents (Wang et al., 2024), which use MLLMs to reason over smaller, fixed length segments of a video that are retrieved based on captions generated by CLIP (Radford et al., 2021) like models or using external tools. However, many such systems (Wang et al., 2025b; Fei et al., 2024; Ranasinghe et al., 2025) rely on predefined reasoning structures and fixed captioning tools that may miss details relevant to a given complicated query's context. By analyzing captions of short, fixed clips in relative isolation, they often miss the broader narrative structure, limiting their ability to perform true long-horizon reasoning. Additionally, reliance on external modules ((Pang and Wang, 2025; Zhang et al., 2025)) can also obscure whether performance gains stem from the agentic reasoning process or the inherent power of the external tools themselves. VideoMind circumvents these limitations by introducing a suite of internal reasoning modes that use the base MLLM's capabilities for multi-granular temporal and spatial analysis.

Appendix A provides a more detailed overview of work related to our method.

## 3 Method

### 3.1 Setup and Notation

Let a long-form video be a sequence of $N$ frames, $V = \{F_1, F_2, \ldots, F_N\}$. Given a question $Q$, the objective is to generate a correct answer $A$.

Our framework is built upon a pre-trained Multimodal Large Language Model (MLLM), denoted as $f_\theta$, where $\theta$ are its parameters. This model processes visual information through a corresponding Vision Transformer (ViT), $g_\phi$, which embeds raw frames. We use $\oplus$ to denote the concatenation of multimodal sequences (text and visual embeddings) that form the input to $f_\theta$.

The reasoning process is a multi-step interaction. At each step $t$, the agent maintains a history of its previous interactions, $H_{t-1} = \{(Z_1, O_1), \ldots, (Z_{t-1}, O_{t-1})\}$. This history is a list of tuples, where $Z_i$ is the agent's textual reasoning and $O_i$ is the structured output from that step. Based on this history, the agent generates a new thought $Z_t$ and selects a reasoning mode $M_t$ along with its configuration, or terminates by producing the final answer $A$.

### 3.2 Framework Overview

As illustrated in Figure 1, Video Mind uses the MLLM $f_\theta$ conditioned on a system prompt, $P_{agent}$

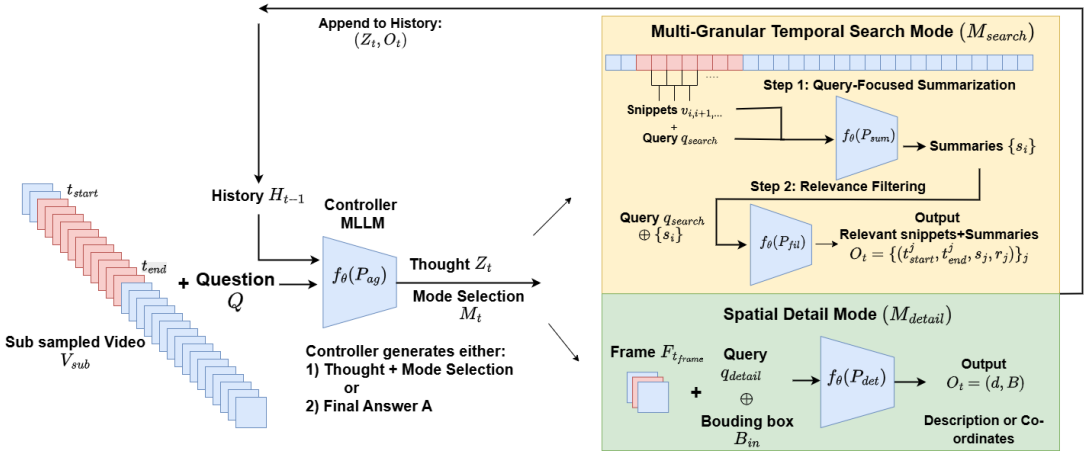


Figure 1: **An overview of Video Mind.** Given a long video and a question, the MLLM agent ($f_\theta$) processes a sparse set of initial frames ($V_{sub}$) and the question $Q$. It then enters a reasoning loop to interact with the video, generating a thought $Z_t$ and selecting a mode $M_t$ along with the configuration details needed. The activated reasoning mode, an instance of $f_\theta$ itself, processes the input to produce an output $O_t$. This output is appended to the history $H_t$, which informs the next reasoning step. The process terminates when the agent has gathered sufficient evidence to produce the final answer $A$.

as the agent orchestrator.

The agent is initialized with the question $Q$ and an initial, condensed view of the video, $V_{sub} \subset V$ consisting of $K$ very sparsely, uniformly sampled frames, providing a coarse overview.

The agent's task is to decompose the complex question $Q$ into a series of simpler sub-problems and solve these sub-problems by interacting with the video through specialized 'modes' of the MLLM (detailed in Section 3.3). This dynamic execution process (detailed in Section 3.4) is iterative: the agent plans a step (thought $Z_t$), engages a mode (selection $M_t$ along with its configuration) to interact with a part of the video, receives new information (the mode's output $O_t$), and updates its history. This loop continues until the agent has gathered sufficient evidence to produce the final answer $A$.

### 3.3 MLLM Modes

Our framework's core consists of two specialized modes that are engaged by the MLLM to interact with the video. Both modes are instantiated from the base MLLM $f_\theta$ only using distinct system prompts.

#### 3.3.1 Multi-Scale Temporal Search Mode

**Purpose and Motivation.** To efficiently navigate long videos and pinpoint temporally relevant events, we design the Multi Scale Temporal search mode $M_{search}$ .

**Inputs and Process.** In this mode, the MLLM requires the following configuration details generated by the orchestrator: a natural language sub-query $q_{search}$, a time interval $[t_{start}, t_{end}]$, and a search granularity scale $\Delta t$ (chosen from $N_{scale}$ fixed options for it). Its operation unfolds in two stages:

**(1) Query-Focused Summarization:** In this stage, a utility partitions the video segment $V_{[t_{start},t_{end}]}$ into non-overlapping snippets $\{v_i\}$ of duration $\Delta t$. From each snippet $v_i$, a maximum of $K_{clip}$ frames are uniformly sampled. These sparse frames are then processed by $f_\theta$ conditioned on a summarization prompt $P_{sum}$ and $q_{search}$ to generate a concise query specific summary $s_i$ for each snippet.

$$s_i = f_\theta(P_{sum} \oplus q_{search} \oplus g_\phi(\{F_j\}_{j \in v_i}))$$

**(2) Relevance Filtering:** In this stage, the complete set of generated summaries $\{s_i\}$ is aggregated. The MLLM $f_\theta$ is invoked again with a filtering prompt $P_{fil}$ to analyze these summaries and identify which snippets are most relevant to the sub-query $q_{search}$.

**Outputs.** Based on the filtering stage, the MLLM constructs its final output, $O_{search}$ a structured list of tuples, where each tuple contains the start and end timestamps of a relevant snippet $v_j$, their summary $s_j$ and a textual justification $r_j$ for its selec-

tion.

$$O_{search} = \{(t^j_{start}, t^j_{end}, s_j, r_j)\}_j$$
$$= f_\theta(P_{fil} \oplus q_{search} \oplus \{s_1, s_2, \dots\})$$

The design of $M_{search}$ directly facilitates a hierarchical, coarse-to-fine search strategy. The agent can first perform a coarse search over a long duration with a large $\Delta t$ to identify broad events of interest. Subsequently, it can "zoom in" by invoking the tool again on the identified relevant segments with a smaller $\Delta t$ for more precise temporal localization.

#### 3.3.2 Spatial Detail Mode

The spatial detail mode $M_{detail}$ is designed to help the agent perform fine-grained spatial analysis to understand specific objects, interactions, or details within a frame after appropriate temporal localization using $T_{search}$.

**Inputs and Process.** Upon selecting the mode $M_{detail}$, the orchestrator generates a configuration consisting of a specific timestamp $t_{frame}$ and a detailed natural language query $q_{detail}$ describing the attribute or object of focus. Let $F_{t_{frame}}$ be the frame at the given timestamp. A key feature of this tool is its ability to focus on specific spatial regions. To support this, the agent can optionally provide bounding box coordinates $B_{in} = (x, y, w, h)$ as an additional input. When $B_{in}$ is provided, a utility first crops the frame, $F_{crop} = \text{crop}(F_{t_{frame}}, B_{in})$. The MLLM then engages this mode with the system prompt $P_{det}$, $q_{detail}$ and $F_{crop}$ to analyze the frame's content in relation to the query, while also localizing the said content by providing bounding box co-ordinates for it. If no $B_{in}$ is provided, the MLLM receives the full frame $F_{t_{frame}}$ instead. $T_{detail}$ helps focus the MLLM's attention on specific spatial regions, mitigating distractions from irrelevant background content.

This input-output design directly facilitates a hierarchical, coarse-to-fine spatial analysis. As described below, the MLLM in this mode can output bounding boxes. The agent can then "zoom in" by invoking $T_{detail}$ again with a more specific subquery (e.g focus on some specific attributes of the object localized in the previous iteration), using the previous iterations outputted box $B$ as the new input box $B_{in}$ for the next iteration.

**Outputs.** The MLLM in mode $M_{detail}$ produces a structured output $O_{detail}$ containing a textual description $d$ that answers the query, and optionally, bounding box coordinates $B$ identifying a specific object or region mentioned in the description.

$$O_{detail} = (d, B)$$
$$= f_\theta(P_{det} \oplus q_{detail} \oplus g_\phi(F_{t_{frame}}))$$

### 3.4 Dynamic Mode Selection and Execution

The controller MLLM, $f_\theta$ with prompt $P_{agent}$, orchestrates the entire process in an iterative, closed-loop manner, as illustrated in Figure 1. At each step $t = 1, \dots, T_{max}$:

**Reasoning and Planning.** The agent model $f_\theta, P_{agent}$ receives the accumulated history $H_{t-1}$, sub-sampled video frames $V_{sub}$, and the question $Q$. It first generates a textual thought $Z_t$ that outlines its reasoning process and articulates a plan for the next action based on the evidence gathered so far.

**Engaging a Mode.** Based on the plan formulated in $Z_t$, the agent initiates a mode selection $M_t$. This selection is formatted in a structured syntax, specifying which mode to engage, $\{M_{search}, M_{detail}\}$, and providing the necessary configuration details. For instance, a selection of the temporal search mode would be $C_t = (M_{search}, \{q_{search}, t_{start}, t_{end}, \Delta t\})$. If the agent determines it has sufficient evidence to answer the question, it can instead generate the final answer $A$ and terminate the loop.

**Mode Execution and History Update.** The MLLM engages the selected mode $M_t$ in its given configuration and executes it with corresponding input $I_t$ to produce the output $O_t$. This new information is then used to update the history by appending the latest reasoning step and model output: $H_t = H_{t-1} \oplus (Z_t, O_t)$. This iterative cycle continues until the agent produces a final answer, allowing it to dynamically adapt its strategy based on the information gathered at each step.

## 4 Experiments and Results

### 4.1 Experimental Setup

**Benchmarks and Metric.** We evaluate our method on three standard long video benchmarks: VideoMME (Fu et al., 2025), LongVideoBench (Wu et al., 2024), and MLVU (Cui et al., 2024).

**Models and Baselines.** We implement VideoMind framework using two open source MLLMs as backbones: Qwen 2.5 72B (Bai et al., 2025)

| Method | Params | LongVideo Bench | VideoMME | | | | MLVU |
|---|---|---|---|---|---|---|---|
| | | Overall | Overall | Short | Medium | Long | Overall |
| GPT-4o (OpenAI, 2023) | - | 66.7 | 71.9 | - | - | 65.3 | 64.6 |
| Gemini-1.5-Pro (Gemini et al., 2024) | - | 64.0 | 75.0 | - | - | 67.4 | 64.0 |
| InternVL2.5 72B (Chen et al., 2024) | 78B | 63.6 | 72.1 | - | - | 62.6 | 75.7 |
| LLaVA-OneVision (Li et al., 2024a) | 72B | 61.3 | 66.2 | - | - | - | 68.0 |
| Qwen2.5-VL+AdaReTaKe (Ma et al., 2024) | 72B | 67.0 | 73.5 | - | - | 65.0 | 78.1 |
| Base Model (Llama 4 Scout) | 17B | 49.5 | 62.8 | 76.5 | 67.1 | 54.2 | 67.4 |
| **VideoMind (Llama 4 Scout)** | **17B** | **53.1** | **67.8** | **79.7** | **76.7** | **59.5** | **73.6** |
| Base Model (Qwen-2.5-VL 72B) | 72B | 60.5 | 72.8 | **82.1** | 70.1 | 60.2 | 74.6 |
| **VideoMind (Qwen-2.5-VL 72B)** | **72B** | **63.1** | **77.6** | 81.5 | **77.8** | **64.5** | **77.2** |

Table 1: **Video Mind on long video understanding benchmarks.** We compare the performance of **VideoMind**, against baselines using two backbones: Llama 4 Scout and Qwen-2.5-VL (72B).

and Llama 4 Scout (AI, 2025) (17B). We note that smaller models (e.g., 7B parameters) were found to be unsuitable. They lacked the sufficient instruction-following capabilities required by our framework, e.g. given input video frames they struggled to generate a structured mode selection configuration like the one in Sec. 3.4. We compare VideoMind against its corresponding Base Model version. For this baseline, the MLLM is given 768 uniformly sampled frames and the user query directly. We also compare our models against recent LVU baselines and closed source methods to contextualize the performance improvements achieved by VideoMind.

**Implementation Details.** VideoMind is initialized with a sparse, uniformly sampled set of 64 frames ($V_{sub}$). We provide 1-shot examples of mode selection in the prompt. We use a $N_{scale} = 5$ time scale options ($10, 30, 90, 270, 600$ seconds) in our Multi Scale temporal search mode. To ensure a fair evaluation and prevent data leakage, these examples are drawn from a separate dataset (e.g., using MLVU examples when evaluating on VideoMME). Across all experiments, the agent is limited to engage in modes a maximum of 20 times per question.

### 4.2 Long Video Understanding Performance

As shown in Table 1, our reasoning mode based approach provides consistent and significant gains across all benchmarks. On VideoMME, VideoMind improves the Llama 4 Scout accuracy from 62.8% to 67.8% and the Qwen 2.5 72B from 72.8% to 77.6%. The largest gains are observed on Medium and Long videos, where temporal localization is most critical. For instance, Llama 4 Scout improves from 67.1% to 76.7% on Medium videos and 54.2% to 59.5% on Long videos. Similar improvements over the base model are observed on LongVideoBench and MLVU, as detailed in Table 1, demonstrating the general applicability of our framework.

### 4.3 Which kind of tasks benefit the most?

To understand the specific capabilities enhanced by our framework, we conduct a category-wise breakdown of performance on Video-MME and MLVU in Figure 2, using Llama 4 Scout.

On Video-MME (Fig. 2 left), VideoMind provides substantial improvements on tasks requiring complex temporal understanding. The most significant gains are in Action Reasoning (e.g., from 42% to 68%), Temporal Reasoning (from 50% to 65%), and Temporal Perception (from 52% to 68%). This highlights the effectiveness of our Multi-Scale Temporal Search mode in guiding the MLLM to identify and process the most salient events across different times.

A similar trend is visible on the MLVU benchmark in Fig. 2 (right). Our framework shows the largest improvements on Action Order (e.g., from 52% to 70%) and PlotQA (from 65% to 75%), both of which require synthesizing information from multiple, distinct moments in the video.

### 4.4 Ablation Studies

#### 4.4.1 Individual Mode Contributions

To understand the individual contributions of our two modes, we conduct an ablation study where we restrict the modes available to the agent (Table 2). We compare four settings: (1) the Base Model with no modes (2) $M_{detail}$ only, (3) $M_{search}$ only, and (4) our Full Method (VideoMind) with both modes.

The results reveal a clear trend. Providing only the Spatial Detail mode (+ $M_{detail}$ only) offers only

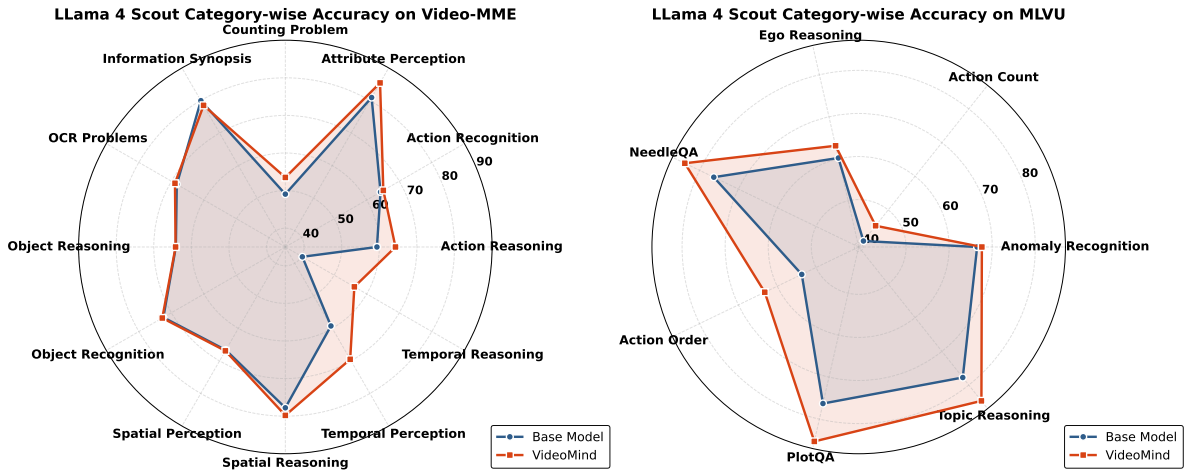


Figure 2: Category-wise comparison of VideoMind's performance compared to the base model using the Llama 4 Scout backbone on Video MME (left) and MLVU (right)

| Method | VideoMME | | LongVideoBench |
|---|---|---|---|
| | Long | Overall | Overall |
| Base Model | 54.2 | 62.8 | 49.5 |
| + $M_{detail}$ only | 54.2 | 64.5 | 50.1 |
| + $M_{search}$ only | 58.7 | 66.9 | 52.4 |
| **VideoMind** | **59.5** | **67.8** | **53.1** |

Table 2: **Ablation on the individual contributions of the Temporal Search ($M_{search}$) and Spatial Detail ($M_{detail}$) modes.** All results are for Llama 4 Scout.

a modest improvement over the baseline (64.5% vs. 62.8% on VideoMME Overall), as the MLLM struggles to locate the correct frames to analyze in longer videos. Conversely, providing only the Temporal Search mode (+ $M_{search}$ only) captures the vast majority of the performance gain, achieving 66.9% Overall. This demonstrates that effective temporal localization is the most critical factor for long video understanding. Nonetheless, using both modes for fine-grained analysis, achieves the highest performance (67.8%), confirming that $M_{detail}$ provides a valuable, complementary benefit.

#### 4.4.2 Temporal Mode Design: Granularity Levels

| Granularity Lvl. | VideoMME | | LongVideoBench |
|---|---|---|---|
| | Long | Overall | Overall |
| 2 Levels | 57.7 | 67.1 | 52.4 |
| 3 Levels | 59.2 | 67.6 | 53.0 |
| **5 Levels** | **59.5** | **67.8** | **53.1** |

Table 3: **Ablation on the number of time scale levels available to the Temporal Search Mode ($M_{search}$).** All results are for Llama 4 Scout.

We now ablate the design of our Multi-Scale Temporal Search Mode ($M_{search}$), specifically the impact of the number of available options for the time scale $\Delta t$. We compare our 5 level method against versions with 2 levels $(10, 270$ s) and 3 levels $(10, 90, 270$ s).

As detailed in Table 3, increasing the levels from 2 to 3 provides a clear boost from 57.7% to 59.5% on VideoMME Long subset and 52.4% to 53.0% on LongVideoBench. However, we observe diminishing returns beyond this point. Increasing to 5 levels yields only marginal gains (0.3% on VideoMME Long and 0.1% on LongVideoBench).

#### 4.4.3 Visual Detail Mode Design

| Method | Video MME | LongVideo Overall | LongVideo S2A |
|---|---|---|---|
| **VideoMind** | **67.8** | **53.5** | **72.5** |
| w/ Grounding DINO | 56.1 | 44.6 | 56.1 |
| w/o Crop | 66.4 | 52.2 | 64.3 |

Table 4: **Ablation on the Spatial Detail Mode ($M_{detail}$) design.** Our full method VideoMind is compared against alternatives. Evaluated on Llama 4 Scout.

We first analyze the design of our Spatial Detail mode ($M_{detail}$) in Table 4. Our full method empowers the MLLM to perform iterative, focused visual inspection by cropping and zooming. We compare this against two alternatives: (1) replacing $M_{detail}$ with an open-vocabulary detector (GroundingDINO) that operates with the same input parameters and (2) a simplified version of the mode ('w/o Crop') where the mode can only view the full frame.

The results confirm our design. The detector-based approach yields significantly worse performance (56.1% vs. 67.8% on VideoMME Overall), as it struggles with the open-ended nature of the

queries. More importantly, removing the crop-and-zoom capability ('w/o Crop') slightly degrades performance from 67.8% to 66.4% on VideoMME and from 53.5% to 52.2% on LongVideo. The performance gap is most pronounced on the LongVideo S2A subset, dropping from 72.5% to 64.3%, which specifically requires fine-grained spatial attribute recognition. This validates our design that enables the MLLM to perform focused, iterative visual inspection.

## 5 Conclusion

We introduce VideoMind, a novel, training-free agentic framework where an MLLM mimics the human reasoning process for long video understanding. It overcomes fixed context limits by decomposing queries into multi-step plans and executing them by transitioning into specialized internal reasoning modes for multi-granular temporal search and spatial detail analysis. This zero-shot approach yields substantial gains on benchmarks like Video-MME (improvement of 4.8% to 77.6% when using Qwen 2.5 VL 72B), demonstrating that complex reasoning can be unlocked from base MLLMs intrinsic reconfiguration without costly retraining. While performance is bound by the MLLM's fidelity, this scalable, self-specialization paradigm represents a promising direction for long-form video analysis.

## 6 Limitations

The performance of VideoMind is fundamentally bound by the capabilities of the underlying MLLM. Its effectiveness relies on the model's reasoning and instruction-following fidelity to both orchestrate the investigative plan and manage transitions between reasoning modes effectively. Consequently, our framework requires a sufficiently powerful base model. As noted in our experiments, smaller models (e.g., 7B parameters) were found to be unsuitable, as they lacked the necessary instruction-following capabilities to manage the iterative mode-switching workflow. Furthermore, the multi-step nature of the reasoning loop requiring a forward pass of the model in each step may introduce additional computational overhead, potentially increasing total inference latency depending on the number of mode engagements required.

Additionally, our current approach is training-free, relying on 1-shot examples to guide mode engagement. While this demonstrates strong zero-shot generalization, the agent's planning strategy is not explicitly optimized. Future work could explore lightweight fine-tuning to further improve performance, as an optimized strategy for selecting and utilizing these internal modes may yield further gains.

## References


Meta AI. 2025. The llama 4 herd: The beginning of a new era of natively multimodal ai innovation. *https://ai. meta. com/blog/llama-4-multimodal-intelligence/, checked on*, 4(7):2025.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, and Chang Zhou. 2023. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and more. *arXiv preprint arXiv:2308.12966*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Ilia Bulatov and Victor Lempitsky. 2022. Recurrent model of visual attention for analyzing long videos. *arXiv preprint arXiv:2204.05802*.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, and 1 others. 2024. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*.

Meng Chu, Yicong Li, and Tat-Seng Chua. 2025. Understanding long videos via llm-powered entity relation graphs. *arXiv preprint arXiv:2501.15953*.

Zongheng Cui, Jian Liang, Peixian Wang, Jie Huang, Jun Xiao, and Han Zhang. 2024. Mlvu: A multi-level long video understanding benchmark for large vision language models. *arXiv preprint arXiv:2405.19534*.

Alexey Dosovitskiy. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Hao Fei, Shengqiong Wu, Wei Ji, Hanwang Zhang, Meishan Zhang, Mong-Li Lee, and Wynne Hsu. 2024. Video-of-thought: Step-by-step video reasoning from perception to cognition. *arXiv preprint arXiv:2501.03230*.

Chaoyou Fu, Guoli Chen, Yixuan Yin, Xinyu Wang, Jincan Ye, Zheyuan Lin, Yikang Li, and Howard Luo. 2024. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu

Zhou, Yunhang Shen, Mengdan Zhang, and 1 others. 2025. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24108–24118.

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

Kristen Grauman, Andrew Westbury, Eugene Byrne, C Chavis, A Furnari, R Girdhar, J Hamburger, H Hao, D Hendricks, S Jandial, and 1 others. 2022. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18995–19012.

Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang, Jianyu Jiang, Jiawei Wang, and 1 others. 2025. Seed1. 5-vl technical report. *arXiv preprint arXiv:2505.07062*.

Hongji Guo, Hanjing Wang, and Qiang Ji. 2022. Uncertainty-guided probabilistic transformer for complex action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19968–19978.

Yixin He, Siyuan Song, and Min Xu. 2024. Video-rag: A training-free framework for detail-oriented long-video understanding. *arXiv preprint arXiv:2405.02101*.

Jindong Jiang, Xiuyu Li, Zhijian Liu, Muyang Li, Guo Chen, Zhiqi Li, De-An Huang, Guilin Liu, Zhiding Yu, Kurt Keutzer, and 1 others. 2025. Storm: Token-efficient long video understanding for multimodal llms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5830–5841.

Min Jung Lee, Dayoung Gong, and Minsu Cho. 2025. Video summarization with large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 18981–18991.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and 1 others. 2024a. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*.

Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, Limin Wang, and Yu Qiao. 2024b. Mvbench: A comprehensive multi-modal video understanding benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22195–22206.

Hao Liu, Matei Zaharia, and Pieter Abbeel. 2024a. Ringattention with blockwise transformers for near-infinite context. In *The Twelfth International Conference on Learning Representations*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*.

Yuanxin Liu, Shicheng Li, Yi Liu, Yuxiang Wang, Shuhuai Ren, Lei Li, Sishuo Chen, Xu Sun, and Lu Hou. 2024b. TempCompass: Do video LLMs really understand videos? In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 8731–8772, Bangkok, Thailand. Association for Computational Linguistics.

Hongxiao Ma, Ziyang Chen, Ya-chu Wang, Ao Fan, and Jing Yang. 2024. Adaretake: A novel approach for long-video understanding. *arXiv preprint arXiv:2406.11029*.

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

OpenAI. 2023. Gpt-4v (ision) system card. Accessed: 2024-10-07.

Ziqi Pang and Yu-Xiong Wang. 2025. MR. video: Mapreduce as an effective principle for long video understanding. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Jinyoung Park, Hee-Seon Kim, Kangwook Ko, Minbeom Kim, and Changick Kim. 2024. Videomamba: Spatio-temporal selective state space model. In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part XXV*, page 1–18, Berlin, Heidelberg. Springer-Verlag.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR.

Kanchana Ranasinghe, Xiang Li, Kumara Kahatapitiya, and Michael S Ryoo. 2025. Understanding long videos with multimodal language models. In *The Thirteenth International Conference on Learning Representations*.

Luca Rossetto, George Awad, Werner Bailer, Cathal Gurrin, Björn Jónsson, Jakub Lokoč, Stevan Rudinac, and Klaus Schoeffmann. 2025. Overview of the 1st international workshop on interactive video search and exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessi, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *arXiv preprint arXiv:2302.04761*.

Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. 2020. Finegym: A hierarchical video dataset for fine-grained action understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2616–2625.

Xiaoqian Shen, Yunyang Xiong, Changsheng Zhao, Lemeng Wu, Jun Chen, Chenchen Zhu, Zechun Liu, Fanyi Xiao, Balakrishnan Varadarajan, Florian Bordes, and 1 others. 2024. Longvu: Spatiotemporal adaptive compression for long video-language understanding. *arXiv preprint arXiv:2410.17434*.

Hantao Song, Yitong Zhang, Ming Song, Zelin Li, and Han Xu. 2024. Moviechat: From dense token to sparse memory for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18485–18495.

Gemini Team. 2023. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Yifei Tian, Zhipeng Zhang, Kevin Li, Zihan Yao, Zhiqiang Zhang, Zihan Zeng, Lu Jiang, Fei-Fei Li, and Min Xu. 2024. Ego-r1: A million-scale egocentric video benchmark for reasoning about object interactions. *arXiv preprint arXiv:2405.15582*.

Weihan Wang, Zehai He, Wenyi Hong, Yean Cheng, Xiaohan Zhang, Ji Qi, Ming Ding, Xiaotao Gu, Shiyu Huang, Bin Xu, and 1 others. 2025a. Lvbench: An extreme long video understanding benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22958–22967.

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

Ziyang Wang, Shoubin Yu, Elias Stengel-Eskin, Jaehong Yoon, Feng Cheng, Gedas Bertasius, and Mohit Bansal. 2025b. Videotree: Adaptive tree-based video representation for llm reasoning on long videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3272–3283.

Yuetian Weng, Mingfei Han, Haoyu He, Xiaojun Chang, and Bohan Zhuang. 2024. Longvlm: Efficient long video understanding via large language models. In *European Conference on Computer Vision*, pages 453–470. Springer.

Chao-Yuan Wu, Georgia Gkioxari, Christoph Feichtenhofer, Ross Girshick, and Kaiming He. 2022. Memvit: Memory-augmented multiscale vision transformer for efficient long-term video recognition. In *European Conference on Computer Vision*, pages 192–209. Springer.

Yuxuan Wu, Ziyi Wang, Chen Bai, Zhaoxiang Zhang, and Chen Zhu. 2024. Longvideo: A benchmark for assessing long-video understanding capabilities of video large language models. *arXiv preprint arXiv:2406.01438*.

Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. 2021. Next-qa: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9777–9786.

Shoubin Yu, Jaemin Cho, Prateek Yadav, and Mohit Bansal. 2023. Self-chained image-language model for video localization and question answering. *Advances in Neural Information Processing Systems*, 36:76749–76771.

Xiaoyi Zhang, Zhaoyang Jia, Zongyu Guo, Jiahao Li, Bin Li, Houqiang Li, and Yan Lu. 2025. Deep video discovery: Agentic search with tool use for long-form video understanding. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

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

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, Zhangwei Gao, Erfei Cui, Xuehui Wang, Yue Cao, Yangzhou Liu, Xingguang Wei, Hongjie Zhang, Haomin Wang, Weiye Xu, and 32 others. 2025. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *Preprint*, arXiv:2504.10479.

# Supplementary Material: VideoMind: Thinking in Steps for Long Video Understanding

## A Related Work

Our work, VideoMind, builds upon advancements in Multimodal Large Language Models (MLLMs), Long Video Understanding (LVU), and the emerging paradigm of agentic AI systems that leverage external tools.

### A.1 Multimodal Large Language Models

The field of multimodal AI has been revolutionized by MLLMs (OpenAI, 2023; Liu et al., 2023; Bai et al., 2023; Team, 2023; AI, 2025; Zhang et al., 2024; Zhu et al., 2025; Guo et al., 2025), which integrate vision encoders with powerful Large Language Models (LLMs) demonstrating remarkable zero-shot capabilities in a wide range of vision-language tasks. These models typically employ a vision encoder (e.g., ViT Dosovitskiy (2020)) to extract image features, which are then mapped into the LLM's token space via a projection module. While highly effective for static images and short video clips, these architectures face significant limitations when applied to long-form video. The primary bottlenecks are the quadratic complexity of self-attention mechanisms and fixed context windows, which make processing the vast number of tokens from hour-long videos computationally prohibitive and prone to information loss or catastrophic forgetting (Fu et al., 2025). Our work circumvents this by using the MLLM not as a passive processor of all frames, but as an intelligent agent that actively seeks relevant information.

### A.2 Long Video Understanding

**Benchmarks.** Early benchmarks like Ego4D (Grauman et al., 2022) provided large-scale datasets with long, egocentric videos, focusing on episodic memory. More recent benchmarks have been designed to test the long-context reasoning capabilities of MLLMs across a spectrum of tasks. These include comprehensive evaluations like Video-MME (Fu et al., 2024) and the 20-task MVBench (Li et al., 2024b). Others focus on extreme length and "needle-in-a-haystack" retrieval, such as LongVideo (Wu et al., 2024) with its hour-long content, and LVBench (Wang et al., 2025a) which uses TV series and sports. Specialized benchmarks probe deeper temporal reasoning: EgoSchema (Mangalam et al., 2023) targets long-form reasoning where answers are not localized to short clips, NExT-QA (Xiao et al., 2021) focuses on causal and temporal action rationale, and TempCompass (Liu et al., 2024b) provides a fine-grained evaluation of temporal order. In this work, we focus on three diverse benchmarks to demonstrate generalizability: Video-MME (Fu et al., 2025), MLVU (Cui et al., 2024), and LongVideo(Wu et al., 2024).

**Approaches for LVU.** Existing approaches to LVU can be broadly categorized by how they manage the computational burden of long contexts. One line of work focuses on efficient architectures. This includes models with recurrent state management like Bulatov and Lempitsky (2022), adopting linear-complexity State Space Models (SSMs) like VideoMamba (Park et al., 2024), or using efficient attention mechanisms like RingAttention (Liu et al., 2024a) to scale to millions of tokens. A second strategy involves token reduction or compression. This ranges from compressed memory banks like MemViT (Wu et al., 2022) and segment-based feature aggregation like LongVLM (Weng et al., 2024), to more advanced Mamba-based temporal projectors like STORM (Jiang et al., 2025) that merge spatiotemporal information. A third category employs coarse-to-fine or retrieval-based mechanisms. For instance, SeViLA (Yu et al., 2023) employs a language-aware localizer to retrieve keyframes, while AdaRetake (Ma et al., 2024) learns to "retake" relevant segments from a memory bank.

More recently, this retrieval-based philosophy has recently evolved into agentic tool use frameworks. Systems like Toolformer (Schick et al., 2023) demonstrated that LLMs can learn to use external APIs, and this paradigm has been extended into agentic frameworks that perform complex reasoning by planning and executing actions (**?**Ranasinghe et al., 2025). In LVU, this has inspired agentic approaches like VideoAgent (Wang et al., 2024), MovieChat (Song et al., 2024), Video-RAG (He et al., 2024), Graph-VideoAgent (Chu et al., 2025), Ego-R1 (Tian et al., 2024), Deep Video Discovery (Zhang et al., 2025) and MR. Video(Pang and

Wang, 2025). A common strategy in these works is to orchestrate external, specialized models, such as dedicated retrieval systems or powerful closed-source APIs. While effective, this reliance on external modules can obscure whether performance gains stem from the agentic reasoning process or the power of the external tools themselves.

In contrast, our work investigates how a structured, agentic process can unlock the latent capabilities within the base MLLM without reliance on heavyweight external modules. We introduce a novel suite of reasoning modes, including mechanisms for multi-granular temporal search and fine-grained spatial analysis, designed to leverage the MLLM's own intrinsic functions. Our framework, VideoMind, demonstrates that significant improvements can be unlocked from the base MLLM through this structured, zero-shot process. Crucially, it is implemented in a completely training-free manner, making it a lightweight and versatile solution adaptable to various off-the-shelf MLLMs.

## B Robustness to In-Context Example Selection

| Method | LongVideoBench | VideoMME | MLVU |
|---|---|---|---|
| Base Model | 60.5 | 72.8 | 74.6 |
| **VideoMind** | **63.1** $\pm$0.4 | **77.6** $\pm$0.6 | **77.2** $\pm$1.1 |

Table 5: **Robustness of VideoMind to in-context example selection.** We report the average and standard deviation (in parentheses) across 5 independent runs using different randomly sampled in-context examples. Performance remains stable, demonstrating that the selection of our specialization modes are robust to the specific choice of exemplar.

In this section we evaluate the sensitivity of our framework to the specific choice of in-context example used in $P_{agent}$ . We follow the same setup described in in Section 4.1 of the paper for our experiment, using 64 frames ($V_{sub}$) and providing a single example of mode selection and configuration within the system prompt while using Qwen-2.5-VL 72B as the base model for our experiments. As in Section 4.1, these examples are drawn at random from a separate dataset (e.g., utilizing MLVU examples when evaluating on VideoMME).

We find that the specific choice of exemplar has a limited impact on the overall effectiveness of the reasoning loop. As shown in Table 5, **VideoMind** maintains better performance than the Base Model with the standard deviation being much lower than performance improvement across different datasets. The results for our method are reported as the average of 5 independent runs.