

# AgentRVOS: Reasoning Over Object Tracks for Zero-Shot Referring Video Object Segmentation

Woojeong Jin\*, Jaeho Lee\*, Heeseong Shin,  
Seungho Jang, Junhwan Heo, and Seungryoung Kim†

KAIST AI

Project page: <https://cvlab-kaist.github.io/AgentRVOS>

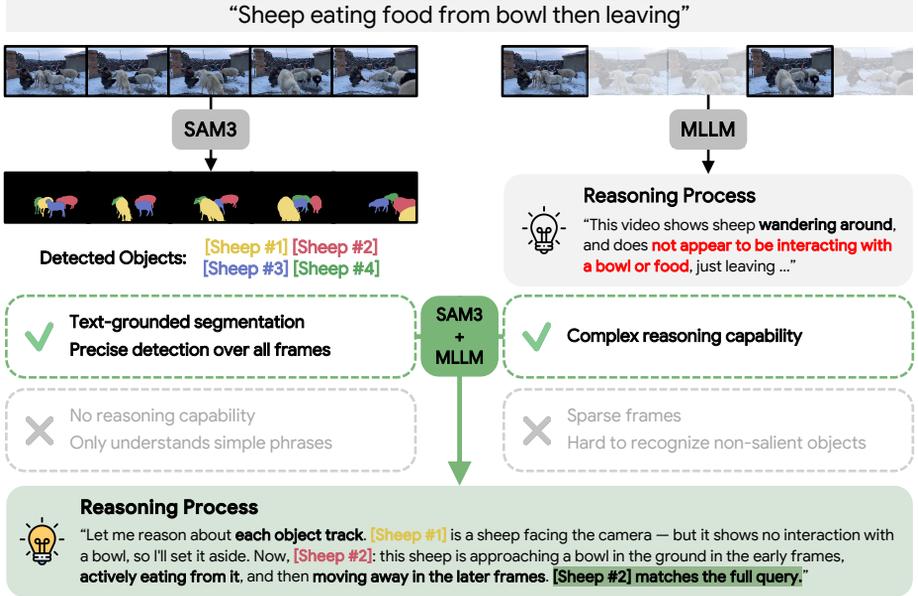
**Abstract.** Referring Video Object Segmentation (RVOS) aims to segment a target object throughout a video given a natural language query. Training-free methods for this task follow a common pipeline: an MLLM selects keyframes, grounds the referred object within those frames, and a video segmentation model propagates the result. While intuitive, this design asks the MLLM to make temporal decisions before any object-level evidence is available, limiting both reasoning quality and spatio-temporal coverage. We propose **AgentRVOS**, a training-free agentic pipeline built on the complementary strengths of SAM3 and an MLLM. Given a concept derived from the query, SAM3 provides reliable perception over the full spatio-temporal extent through generated mask tracks. The MLLM then identifies the target through query-grounded reasoning over this object-level evidence, iteratively pruning guided by SAM3’s temporal existence information. Extensive experiments show that AgentRVOS achieves state-of-the-art performance among training-free methods across multiple benchmarks, with consistent results across diverse MLLM backbones.

**Keywords:** Referring Video Object Segmentation · Agentic AI

## 1 Introduction

Referring Video Object Segmentation (RVOS) requires generating the segmentation mask tracks of the target object throughout a video based on a given natural language query. Unlike image-level referring segmentation [14, 19, 24], RVOS involves queries that go beyond static appearance descriptions and possesses video-specific challenges such as temporal ordering, complex motions, and inter-object relations [6, 13, 38].

The core difficulty of RVOS lies in two intertwined requirements. First, the model must comprehend complex temporal and relational queries, often involving actions, state changes, or relations that distinguish a specific object among many. Resolving such queries demands not only spatial recognition, but also fine-grained reasoning about the object’s identity and its behavior over time.



**Fig. 1: SAM3 and MLLMs have complementary strengths.** SAM3 can precisely identify objects without missing a single frame, but struggles with complex queries. MLLMs, on the other hand, offer strong reasoning capabilities, but operate on sparse frames and struggle with non-salient objects. AgentRVOS combines the advantages of both SAM3 and MLLM, by interleaving the two models in a complementary manner.

Second, the model must ensure dense spatio-temporal coverage across the video: target objects may be small, non-salient, or appear only briefly within a long sequence, requiring the model to exhaustively examine the temporal span rather than relying on sparse observations.

With the recent advances in multimodal large language models (MLLMs) [11, 20, 23, 31], the strong reasoning capabilities of these billion-scale models have shown significant promise for the RVOS task, particularly in comprehending the complex queries given in the task. Existing approaches adopt MLLMs either through task-specific fine-tuning [2, 22, 38, 40], or more recently, in a training-free manner that directly leverages their native multi-modal reasoning capabilities over images and videos.

For the training-free methods [10, 12, 15], a common approach is to first identify a set of video frames that are relevant to a given query, and then perform object grounding on this set. Then, a video segmentation model, such as SAM2 [29], propagates the initial masks across the remaining frames to produce the full segmentation across the entire video. Consequently, these pipelines heavily rely on the MLLM for both temporal frame selection and spatial grounding. However, MLLMs often operate on sparsely sampled frames due to input token limits, resulting in limited temporal coverage. This makes it difficult to detect objects that appear only briefly or intermittently in long videos, suggesting that offloading precise spatio-temporal perception (*i.e.*, temporal object detection) from the

MLLM would allow the model to focus entirely on its primary strength: complex reasoning.

In this paper, we present AgentRVOS, a training-free **agentic** pipeline for RVOS that leverages the reasoning capabilities of MLLMs to their fullest extent by incorporating SAM3 [3] as a complementary perceptual tool. Given a textual prompt, SAM3 can process all frames of a video and produce high quality mask tracks for every matching object instance, without the need for additional inputs such as points or bounding boxes. This allows us to reliably detect small, occluded objects, while also recognizing briefly appearing objects that MLLMs often overlook, as SAM3 can examine the entire video rather than sparse samples. Consequently, we can identify exactly which frames a given object appears in with frame-level precision, enabling us to leverage the MLLM to reason within this spatio-temporally constrained segment.

However, SAM3 alone is not sufficient for RVOS, as it is designed to accept concepts – often given as short noun phrases (*e.g.*, “*person*”, “*red car*”) – as inputs, and tends to struggle with complex queries including temporal or relational comprehension. For instance, given “*the person who stands up after sitting*”, SAM3 can easily locate each person in the video, but cannot determine which one exactly is exhibiting the described behavior. This is where the powerful reasoning capability of MLLMs becomes essential: the MLLM evaluates the candidates generated by SAM3 against the full query, performing the fine-grained temporal and relational reasoning needed to identify the target. In this way, SAM3 and the MLLM complement each other: **SAM3** provides reliable perception over the **full spatio-temporal extent** of the video, while the **MLLM** contributes **comprehensive query-grounded reasoning** over the resultant object-level evidence, as illustrated in Fig. 1.

Translating this complementary structure into practice requires addressing an additional challenge: SAM3’s concept-level candidate generation is intentionally exhaustive, producing a large and diverse candidate pool to ensure high recall. Presenting all candidates to the MLLM at once, however, is impractical, as overlapping mask visualizations may degrade reasoning quality and candidates may span different temporal intervals. We therefore design AgentRVOS as an iterative agentic pipeline that progressively narrows both the candidate set and the temporal scope. At each iteration, MLLM progressively accepts or rejects candidates based on available evidence, while SAM3’s temporal existence information guides focused re-examination of the remaining uncertain cases. Through this progressive narrowing—fewer candidates, tighter temporal windows, and simpler reasoning at each iteration—the pipeline converges on the target object without requiring an explicit frame selection module or exhaustive processing of the entire video.

Our contributions are summarized as follows:

- We propose AgentRVOS, a training-free agentic pipeline for RVOS that combines SAM3’s language-grounded perception with the MLLM’s reasoning capabilities. By delegating object detection and temporal localization to

SAM3, our pipeline allows the MLLM to reason over structured, object-level evidence.

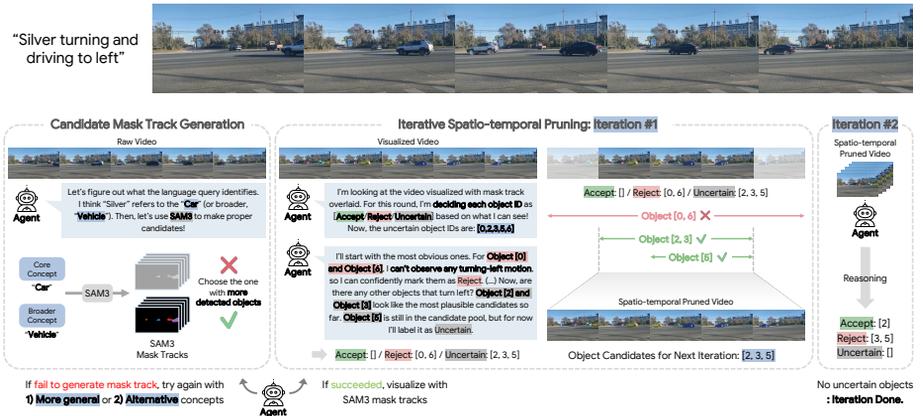
- We introduce an iterative spatio-temporal pruning strategy in which the MLLM progressively eliminates candidates while SAM3’s temporal existence information narrows the relevant temporal scope, decomposing the complex selection problem into progressively simpler reasoning steps.
- Extensive experiments demonstrate that AgentRVOS achieves state-of-the-art performance across multiple benchmarks. Our pipeline consistently shows strong results with various open-source and closed-source MLLMs, demonstrating its generalizability.

## 2 Related Work

**Referring Video Object Segmentation.** RVOS aims to segment target objects in a video based on natural language expressions. Earlier RVOS datasets [7, 16, 30] mainly focused on appearance-based expressions, where objects could be identified through static visual attributes. Consequently, pioneering works [4, 9, 25, 30, 36] demonstrated that query-based architectures—built upon DETR [4]—can effectively link appearance-based textual descriptions with visual representations of objects. However, more recent benchmarks introduce more challenging queries that go beyond appearance-based descriptions. For example, MeViS [6] introduces motion-centric expressions that require temporal reasoning and, ReVOS [38] requires strong reasoning capabilities and world knowledge. These emerging datasets highlight the need for reasoning capabilities beyond simple appearance matching.

**MLLM-based Reasoning Video Object Segmentation.** To enhance reasoning ability for more and more challenging RVOS datasets, recent methods incorporate multimodal large language models (MLLMs) [1, 5, 20, 27] with foundation segmentation models [17, 29] through large-scale supervised fine-tuning [2, 38, 40] or reinforcement learning based optimization [21, 37]. These approaches demonstrate strong performance and improved generalization to out-of-distribution samples.

However, such training-based paradigms typically require substantial annotated data and computational resources. To address these issues, training-free agentic approaches that leverage the zero-shot reasoning capability of MLLMs have recently emerged. For example, CoT-RVS [15] exploits the zero-shot Chain-of-Thought capability of MLLMs to select key frames, applies an image segmentation model to obtain masks for the target object on the selected frames, and propagates them across the video using a video processor. Similarly, ReferAgent [12] leverages CLIP [28] and MLLMs for frame selection and utilizes MLLM-based grounding to prompt segmentation models. Although these methods achieve competitive results without additional training, they rely heavily on the reasoning and grounding capabilities of MLLMs. MLLMs possess strong reasoning capabilities, but they typically process only a limited subset of frames



**Fig. 2: Overall pipeline.** Given a video and a natural language query, our pipeline operates in two phases. In Candidate Mask Track Generation (Sec. 3.2), the MLLM first analyzes the query to extract concepts, which SAM3 uses to produce temporally consistent candidate mask tracks; this process iterates to ensure sufficient coverage. In Iterative Spatio-temporal Pruning (Sec. 3.3), the MLLM reasons over the candidate pool, classifying each candidate as **Accepted**, **Rejected**, or **Uncertain**, while progressively narrowing the spatio-temporal scope until convergence.

due to token limitations. As a result, the model may struggle to perform reasoning centered on the referred object, particularly when the target object appears only sparsely over time. Moreover, the zero-shot grounding capability of MLLMs remains limited when handling small, blurred, or visually ambiguous objects, which can propagate errors across the video. To address this limitation, we introduce AgentRVOS, a training-free agentic pipeline which incorporates SAM3 to provide dense spatio-temporal object-level evidence across the entire video. This allows the MLLM to focus on query-query grounded reasoning over reliable object-level cues.

## 3 Method

### 3.1 Problem Formulation and Overview

Given a video  $\mathcal{V} = \{I_t\}_{t=1}^T$  consisting of  $T$  frames and a natural language query  $Q$ , Referring Video Object Segmentation (RVOS) aims to predict a sequence of  $T$  binary masks  $\mathcal{M} \in \{0, 1\}^{T \times H \times W}$  corresponding to the specified object, where  $H$  and  $W$  denote the spatial dimensions. To effectively tackle this, our approach leverages a synergistic integration of SAM3 [3] and an MLLM. The MLLM enhances SAM3’s capabilities by translating the complex query  $Q$  into a more interpretable prompt, while SAM3 reciprocates by extracting precise spatio-temporal mask tracks. These generated tracks serve as focused visual priors, enabling the MLLM to perform targeted reasoning on specific object candidates rather than exhaustively processing the entire video at once.

Realizing this complementary pipeline, however, requires addressing two core challenges. First, the candidate mask tracks produced by SAM3 must actually contain the referred object; incomplete coverage at this stage cannot be recovered by later reasoning. Second, because SAM3 operates with limited semantic understanding of  $Q$ , necessitating a subsequent reasoning phase that can reliably distinguish the exact target. Our method therefore proceeds in two phases: a candidate generation phase that prioritizes recall to ensure coverage, followed by a spatio-temporal pruning phase in which the MLLM leverages the object-level evidence provided by these tracks to identify and retain only the referred object.

In the first phase, **Candidate Mask Track Generation** (Sec. 3.2), the pipeline constructs a comprehensive pool of object candidates from the video. The MLLM first analyzes the query  $Q$  to determine whether it can be resolved from language alone (*i.e.*, *referring*) or requires visual context from the video (*i.e.*, *reasoning*). Based on this, the MLLM extracts concept-level inputs, especially noun-phrase inputs, and SAM3 then generates temporally consistent candidate mask tracks for each concept. This process iterates to ensure sufficient candidate coverage, expanding to broader or alternative concepts when the initial set is insufficient. In the second phase, **Iterative Spatio-temporal Pruning** (Sec. 3.3), the MLLM performs query-grounded reasoning over the full candidate pool to identify the target. At each iteration, the MLLM classifies candidates as [Accepted, Rejected, Uncertain]. The temporal scope is then narrowed to focus on the frames where uncertain candidates exist, and the pruning repeats until no uncertain candidates remain. A detailed description of AgentRVOS is provided in Appendix A.2– A.4.

### 3.2 Candidate Mask Track Generation

While it would be straightforward to directly predict mask tracks with SAM3 with the given query  $Q$ , SAM3 is trained to segment **concepts** - mainly referring to short noun phrases (*e.g.*, “*person*”, “*red car*”), and tends to struggle with complex queries in RVOS, which requires comprehensive temporal or relational understanding. This motivates us to break down the complex query into simpler and more understandable concepts for SAM3 to handle.

**Concept Extraction.** To this end, we first use the MLLM to pre-process the query  $Q$  and extract a set of concepts  $C$  that SAM3 can handle more reliably. For **referring** queries, where the target itself is self-contained in the query (*e.g.*, “*the cat sitting on the red couch*”), therefore can be identified from language alone without accessing the video. For **reasoning** queries, where the target is defined through temporal or contextual cues that require visual understanding (*e.g.*, “*the one that moves fastest*”), the object itself cannot be inferable solely from the language query. Thus, the MLLM examines sampled video frames alongside the query to infer the relevant object categories.

For extraction, we define two levels of granularity to ensure robustness: **core** concepts and **broader** concepts. **Core** concepts directly correspond to the objects referred to the query, and **broader** concepts, paired with the core concept, are aimed to capture more general categories that can help SAM3 detect instances missed by the core concept. For example, given “*the person who stands up after sitting on the red couch*”, the core concepts would be [person, couch] and the paired broader concepts [human, furniture], respectively.

**Mask Track Generation via SAM3.** With the extracted concepts, we can directly infer SAM3 with the concepts as textual prompts to obtain mask tracks. Between the core and broader concepts from a pair in  $C$ , we select the one that yields more instances from SAM3 to prevent the pipeline from missing out objects, as illustrated in Fig. 2. We collect the mask tracks for each concept pair in  $C$ , yielding a set of mask tracks  $M \in \{0, 1\}^{I \times T \times H \times W}$ , which serve as a pool of **candidate masks**, where  $I$  is the number of instances.

Nonetheless, even with core and broader concepts, SAM3 can still sometimes fail to detect an instance, *i.e.*  $I = 0$ , even with the broader concepts from the extraction stage. In this case, we cascade the extraction process while revising the concepts. Specifically, the subsequent iteration would generate more broadly scoped concepts for referring queries, or exploring alternative object categories for reasoning queries until it is identified by SAM3 in the video.

Each candidate mask  $m_i \in M$  carries two types of information: i) spatial localization through the per-frame segmentation mask, and ii) temporal existence through the set of frames in which the object is present, denoted as  $\mathcal{T}(m_i) = \{t \mid m_i^t \neq \emptyset\}$ , where  $m_i^t$  denotes the binary mask for instance  $i$  at frame  $t$ . This temporal existence information, a natural byproduct of SAM3’s video-level processing, plays a central role in the subsequent pruning phase (Sec. 3.3).

### 3.3 Iterative Spatio-temporal Pruning

Given the generated pool of candidate masks  $M$ , we leverage the complex reasoning capabilities of MLLMs to obtain the query-grounded mask track  $\mathcal{M}$ , which the ability SAM3 lacks. As we have the spatio-temporal masks, we can naturally apply visual prompting [3, 39] on the video to allow the MLLMs to further focus on the objects, as shown in Fig. 2. However, evaluating all candidates in a single pass can be impractical: the pool may contain numerous objects with overlapping masks that would clutter the MLLM, and different candidates may appear at different temporal locations, making a single round of frame sampling insufficient to fairly assess all of them. We therefore adopt an iterative pruning strategy that progressively reduces both the candidate set and the temporal scope, to allow the MLLM to concentrate its reasoning on increasingly fewer, harder cases.

**Candidate Pruning.** At each iteration  $r$ , the MLLM examines the current candidate set  $M^{(r)}$  and classifies each candidate into one of three categories:

**Accepted** (confidently matching the query), **Rejected** (confidently not matching), or **Uncertain** (requiring further evidence). Accepted candidates are collected and merged into the output set  $\mathcal{M} = \bigcup_r \{m_i \in M^{(r)} \mid \text{Accepted}\}$ ; rejected candidates are permanently discarded. Only uncertain candidates carry forward to the next iteration:

$$M^{(r+1)} = \{m_i \in M^{(r)} \mid \text{Uncertain}\}. \quad (1)$$

By committing to confident decisions early, the MLLM avoids repeatedly re-evaluating clear cases and concentrates its reasoning on genuinely ambiguous candidates.

**Temporal Scope Pruning.** As the candidate set shrinks, the relevant temporal scope naturally contracts as well. At iteration  $r$ , the system restricts the temporal scope to the union of frames where the remaining uncertain candidates exist:

$$\mathcal{T}^{(r+1)} = \bigcup_{m_i \in M^{(r+1)}} \mathcal{T}(m_i). \quad (2)$$

Frames are then sampled exclusively within  $\mathcal{T}^{(r+1)}$ , ensuring that every sampled frame contains at least one uncertain candidate.

This achieves two effects simultaneously: it increases the density of informative content in the MLLM’s visual input, and it reduces the total temporal span under consideration. Notably, this mechanism requires no explicit frame selection module—the temporal narrowing emerges naturally from SAM3’s temporal existence information.

**Convergence.** The iterative process terminates when no uncertain candidates remain, *i.e.*,  $M^{(r+1)} = \emptyset$ , meaning all candidates have been either accepted or rejected. In practice, we also impose a maximum iteration count to bound computational cost. Since  $|M^{(r+1)}| \leq |M^{(r)}|$  at every non-trivial iteration, the process is guaranteed to terminate. We observe that most queries converge within a small number of iterations, as the progressive narrowing rapidly reduces ambiguity (see Sec. 4).

## 4 Experiments

**Datasets and Metrics.** We evaluate our method on three major benchmarks for language-guided video object segmentation: MeViS [6], ReVOS [38], and ReasonVOS [2]. These datasets pose distinct challenges. MeViS features complex scenes involving multiple visually similar objects and demands strong motion understanding. ReVOS and ReasonVOS, on the other hand, emphasize reasoning-centric scenarios that require deeper semantic reasoning and world knowledge. Following prior works [2, 15], we report region similarity  $\mathcal{J}$  (average IoU), contour accuracy  $\mathcal{F}$  (mean boundary similarity), and their average  $\mathcal{J}\&\mathcal{F}$ .

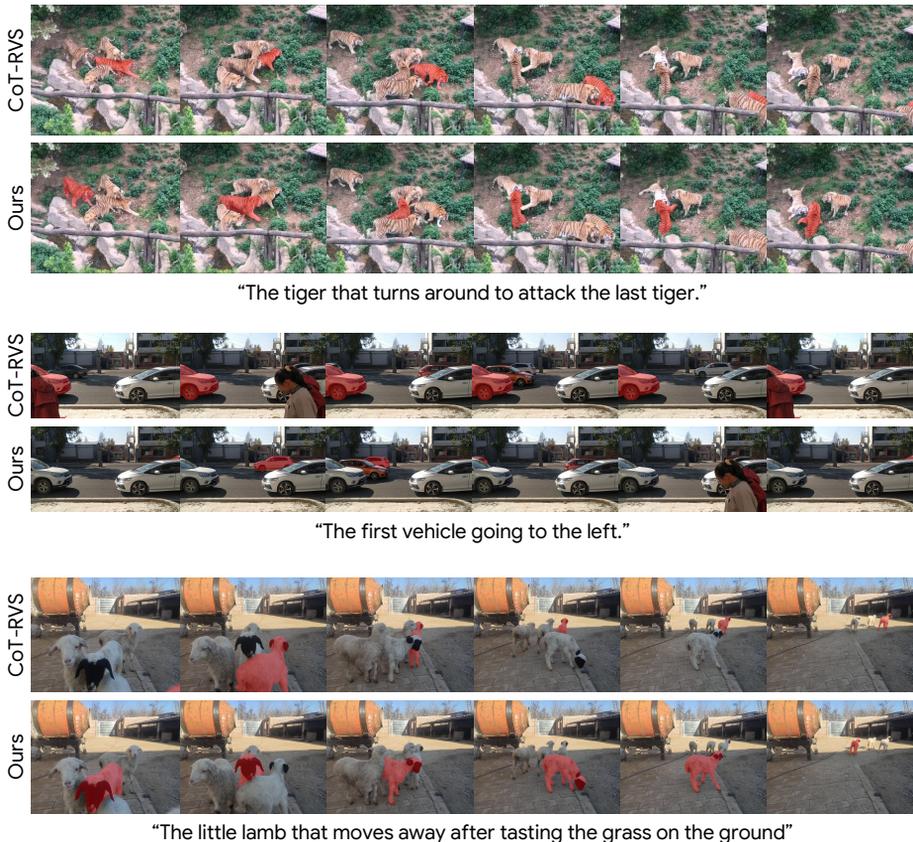
**Table 1:** Comparison with state-of-the-art methods on Referring and Reasoning VOS Benchmarks: MeViS [6], ReVOS [38] and ReasonVOS [2]. Qwen3-VL-8B-T and Qwen3-VL-32B-T indicate Qwen3-VL-8B-Thinking and Qwen3-VL-32B-Thinking model, respectively. The best performing results are presented in **bold**, while the second-best results are underlined. † denotes our reproduced results.

Method	MLLM	MeViS			ReVOS						ReasonVOS					
		$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$	Referring			Reasoning			Overall					
					$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$									
<i>Training-based Methods</i>																
VideolISA [2]	LLaVA-3.8B	41.3	47.6	44.4	-	-	-	-	-	-	-	-	-	45.1	49.9	47.5
VISA [38]	Chat-UniVi-13B	41.8	47.1	44.5	55.6	59.1	57.4	42.0	46.7	44.3	48.8	52.9	50.9	-	-	-
HyperSeg [34]	Mipha-3B	-	-	-	56.0	60.9	58.5	50.2	55.8	53.0	53.1	58.4	55.7	-	-	-
InstructSeg [35]	Mipha-3B	-	-	-	54.8	59.2	57.0	49.2	54.7	51.9	52.0	56.9	54.5	-	-	-
GLUS [22]	LISA-7B	48.5	54.2	51.3	56.0	60.7	58.3	48.8	53.9	51.4	52.4	57.3	54.9	47.5	52.4	49.9
ViLLa [32]	InternVideo2-6B	46.5	52.3	49.4	-	-	-	-	-	-	54.9	59.1	57.0	-	-	-
Sa2VA [40]	InternVL2-8B	-	-	46.9	-	-	-	-	-	-	-	-	57.6	-	-	-
Sa2VA [40]	InternVL3-14B	-	-	-	-	-	-	-	-	-	-	-	60.7	-	-	-
RGA3 [33]	Qwen2.5-VL-7B	47.4	52.8	50.1	58.7	62.3	60.5	53.1	57.7	55.4	55.9	60.0	58.0	51.3	56.0	53.6
VRS-HQ [8]	Chat-UniVi-13B	48.0	53.7	50.9	61.1	65.5	63.3	54.1	59.4	56.8	57.6	62.5	60.0	-	-	-
VideoSeg-R1 [37]	Qwen2.5-VL-7B	52.7	57.8	55.3	-	-	-	-	-	-	58.2	64.0	61.1	-	-	-
<i>Training-free Methods</i>																
AL-Ref-SAM2 [10]	GPT-4	39.5	46.2	42.8	-	-	-	-	-	-	-	-	-	-	-	-
CoT-RVS [15]	Gemma3-12B	40.3	48.1	44.2	-	-	-	-	-	-	43.4	50.9	47.1	47.5	54.0	50.7
CoT-RVS [15]	Qwen3-VL-8B-T†	37.7	43.9	40.8	56.1	61.5	58.8	46.6	53.0	49.8	51.4	57.3	54.3	52.5	58.9	55.7
CoT-RVS [15]	GPT-4o	48.7	55.7	52.2	-	-	-	-	-	-	52.8	59.0	55.9	62.4	68.7	65.5
<b>AgentRVOS (Ours)</b>	Qwen3-VL-8B-T	59.2	64.5	61.9	58.7	62.5	60.6	56.8	61.3	59.0	57.7	61.9	59.8	65.5	71.8	68.6
	Qwen3-VL-32B-T	<u>65.3</u>	<u>70.0</u>	<u>67.7</u>	<u>62.8</u>	<u>66.8</u>	<u>64.8</u>	<u>58.0</u>	<u>62.6</u>	<u>60.3</u>	<u>60.4</u>	<u>64.7</u>	<u>62.5</u>	<u>67.3</u>	<u>73.4</u>	<u>70.4</u>
	GPT-5	<b>70.4</b>	<b>75.7</b>	<b>73.1</b>	<b>66.8</b>	<b>70.8</b>	<b>68.8</b>	<b>61.4</b>	<b>66.0</b>	<b>63.7</b>	<b>64.1</b>	<b>68.4</b>	<b>66.3</b>	<b>73.1</b>	<b>78.0</b>	<b>75.5</b>

**Implementation Details.** We adopt various models as our baseline MLLMs: Qwen3-VL-8B-Thinking [1] and Qwen3-VL-32B-Thinking for open-sourced models, and GPT-5 [26] for closed-source model. As we forementioned, we use additional SAM3 [3] to generate candidate mask tracks. For each video, 16 frames are used by default. The maximum number of iterations is 3 for both concept extraction and iterative spatio-temporal pruning. All experiments are conducted with 4 RTX PRO 5000 Blackwell GPUs. Additional implementation details, including the prompts used in our pipeline, are provided in the supplementary material.

## 4.1 Quantitative Results

Tab. 1 presents quantitative comparisons with state-of-the-art methods on three challenging RVOS benchmarks: MeViS, ReVOS, and ReasonVOS. As shown in the table, AgentRVOS significantly outperforms existing training-free methods achieving up to **40.0%**, **18.6%**, **15.3%** improvements on MeViS, ReVOS, and ReasonVOS respectively. Even when using the same backbone model, AgentRVOS significantly outperforms prior approaches. For instance, with Qwen3-VL-8B-Thinking, AgentRVOS surpasses CoT-RVS [15] using the same model. Moreover, even when compared with pipelines that integrate powerful closed-source models, such as AL-Ref-SAM2 [10] and CoT-RVS with GPT-4o [11], our method still achieves substantially better performance, highlighting the effectiveness of our pipeline design. These results suggest that the spatio-temporal information provided by SAM3 effectively supports MLLMs in both temporal understanding and reasoning over complex video queries.



**Fig. 3: Qualitative results.** AgentRVOS effectively resolves challenging scenarios such as multi-instance ambiguity and temporal reasoning, accurately segmenting the referred objects.

Furthermore, when stronger MLLMs are integrated into our framework, the performance consistently improves. Using Qwen3-VL-32B-Thinking, AgentRVOS achieves state-of-the-art performance across all benchmarks, even compared with training based approaches. Replacing the backbone with strong closed-source models like GPT-5 further improves the results, demonstrating the scalability of our pipeline leveraging stronger MLLM reasoning capabilities.

## 4.2 Qualitative Results

Fig. 3 presents qualitative comparisons between AgentRVOS and CoT-RVS on several challenging referring expressions. In the first example, where multiple instances of the same category appear, AgentRVOS correctly identifies the target by reasoning over inter-object relations, while CoT-RVS fails to distinguish between similar objects. In the second example, our method successfully resolves temporal motion reasoning by identifying the vehicle moving to the left

**Table 2: Component analysis.** We provide ablation study for our core components - (a) retry for the concept extraction in the Candidate Mask Track Generation phase, and (b) the iterative process and (c) temporal scope pruning in the Iterative Spatio-temporal Pruning phase.

Component	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$
(a) w/o Retry in Concept Extraction	65.6	71.8	68.7
(b) w/o Iteration in Iterative Spatio-temporal Pruning	63.0	68.1	65.6
(c) w/o Temporal Scope Pruning	64.2	70.4	67.3
<b>AgentRVOS (Full Pipeline)</b>	<b>67.1</b>	<b>73.6</b>	<b>70.3</b>

among multiple vehicles. In the third example, AgentRVOS accurately performs instance-level reasoning, distinguishing the correct lamb among the same instances. These results demonstrate the effectiveness of AgentRVOS in accurately segmenting the correct objects under challenging scenarios by enabling stronger query-grounded reasoning. Additional qualitative results are provided in the supplementary material.

### 4.3 Analysis

**Component analysis.** In Tab. 2, we provide ablation studies for our core components - (a) retry for the concept extraction in Candidate Mask Track Generation, (b) iteration and (c) temporal scope pruning in Iterative Spatio-temporal Pruning. For concept extraction, we can clearly observe that the (a) retry strategy, play a significant role for preventing the framework from falling into failure cases where SAM3 is not able to detect any objects. We provide detailed analysis of concept extraction in Tab. 5 and in Fig. 5 below.

For the Iterative Spatio-temporal Pruning phase, we can also observe that both the (b) iteration and (c) temporal scope pruning play a significant role, evident from the gains for our full pipeline. This verifies the effectiveness of the core motivation for our framework - MLLMs can perform better reasoning when the model is able to spatio-temporally **focus** in a video, as opposed to existing approaches that naïvely employ the MLLMs to reason the entire video at once, with sampled frames.

#### Ablation on maximum number of iterations for spatio-temporal pruning.

In Tab. 3, we provide quantitative results for varying number of max iterations we apply for Iterative Spatio-temporal Pruning. As shown, we can clearly observe that the iterations show steady improvements, with significant gains up to 3 iterations. As discussed in Sec. 3.3,

we observe that most queries converge in small number of iterations, being 3 in particular, and show minimal improvements with additional iterations further

Max Iteration	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$
1 (No Retry)	63.0	68.1	65.6
2	64.4	70.6	67.5
3	67.1	73.6	70.3
4	67.4	73.7	70.6

**Table 3:** Results for varying maximum number of iterations.



**Fig. 4: Qualitative results of iteration in Iterative Spatio-temporal Pruning.** We illustrate how our iterative spatio-temporal pruning progressively narrows the relevant temporal window and eliminates irrelevant track candidates. Across iterations, the remaining candidates become fewer but more query-consistent, leading to the final selected track set.

on. Therefore, we set our maximum number of iterations as 3, further iterating the process would increase the inference time with marginal improvements.

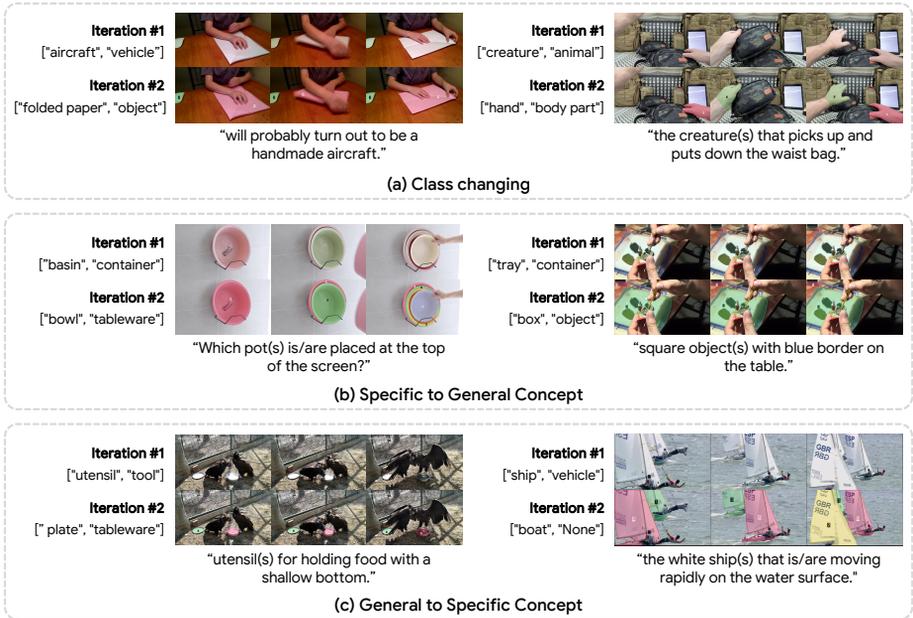
#### Ablation on number of sampled frames.

Our empirical results show that sampling 16 frames yields the best performance of 70.3  $\mathcal{J}\&\mathcal{F}$ . Using fewer frames (8 in this table) limits temporal coverage, while increasing beyond 16 introduces redundancy that may hinder the reasoning ability of the MLLM, with diminishing returns in both accuracy and computational cost. We use 16 frames as the default for all other experiments.

# of Frames	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$
8	64.6	70.7	67.6
16	67.1	73.6	70.3
32	65.6	72.1	68.8
64	64.9	71.5	68.2

**Table 4:** Effect of the number of sampled frames.

**Effectiveness of iteration for Iterative Spatio-temporal Pruning.** In Fig. 4, we provide detailed visual analysis of the effects of iteration for our Iterative Spatio-temporal Pruning phase. For simple cases, as shown on the top, we can observe that the MLLM is able to confidently classify objects into **Accepted** or **Rejected**, thus not resulting in redundant iterations. In more challenging examples, we can observe that the MLLM is capable of identifying objects that the model is uncertain at first glance. As for the example shown in the middle bottom row, we can observe the framework iteratively rejecting objects and focusing on more confusing objects.



**Fig. 5: Qualitative results of iteration in Concept Extraction.** We visualize how our iterative concept extraction progressively refines object concepts from the query. From top to bottom, the iterative process refines concepts through three distinct patterns: (a) the extracted concept class changes entirely to better match the query, (b) a specific concept is broadened to a more general one, and (c) a vague concept is narrowed to a more precise one, each converging to concepts better aligned with the query’s intent.

**Effectiveness of iteration for concept extraction.** In Tab. 5, we report the ratio of empty masks generated from the Candidate Mask Track Generation phase for different number of iterations applied for the concept extraction process. As a comparison, we also report the ratio of empty masks from CoT-RVS [15], where CoT-RVS leverages CLIP to identify query-relevant frames to reason with the MLLM. We can clearly observe that even without additional iterations, we show significantly lower ratio for having empty masks, demonstrating the effectiveness of our approach with MLLM and SAM3. Further adding iterations to the concept extraction stage minimizes the ratio for having empty masks generated from SAM3.

We further provide detailed visual analysis in Fig. 5, where in Fig. 5(a) the framework is able to identify objects that were not previously identified for the **reasoning** queries (*e.g.* folded paper). Furthermore, we can also observe cases where the subsequent retry processes revise the concept as a more general

Iterations	Empty Mask Ratio (%)
CoT-RVS [15]	12.0
1	4.9
2	4.1 (-0.8)
3	3.8 (-1.1)

**Table 5:** Effects of iterations on the ratio of empty mask prediction (%).

(Fig. 5(b)) or more specific (Fig. 5(c)) concept for the **referring** queries, allowing SAM3 to correctly identify objects that it was previously not able to segment.

## 5 Conclusion

In this paper, we present AgentRVOS, a training-free agentic pipeline that combines SAM3’s language-grounded perception with the MLLM’s reasoning capability for Referring Video Object Segmentation tasks. By leveraging SAM3 for object detection and temporal localization, our pipeline enables the MLLM to perform query-grounded reasoning over structured object-level evidence rather than the entire video directly. To effectively utilize this complementary design, we introduce a spatio-temporal pruning strategy that progressively narrows the candidate set and temporal scope, allowing the MLLM to focus on ambiguous objects and refine its reasoning iteratively. Extensive experiments across multiple RVOS benchmarks and various MLLM backbones demonstrate the effectiveness of AgentRVOS, highlighting the complementary strengths of SAM3’s spatio-temporal perception and MLLM-based reasoning.

## Appendix Overview

The appendix provides additional results and analyses:

- Sec. **A** presents additional experiments and analyses:
  - Ablation on the effect of MLLM reasoning (Sec. **A.1**)
  - Robustness of Concept Extraction (Sec. **A.2**)
  - Details of Visual Prompting (Sec. **A.3**)
  - Detailed Algorithm of AgentRVOS (Sec. **A.4**)
  - Visualization of Overall Reasoning Process of AgentRVOS (Sec. **A.5**)
  - Additional Qualitative Results (Sec. **A.6**)
- Sec. **B** provides implementation details of AgentRVOS:
  - Ablation study configurations (Sec. **B.1**)
  - Detailed prompts used in AgentRVOS (Sec. **B.2**)
- Sec. **C** discusses future directions.

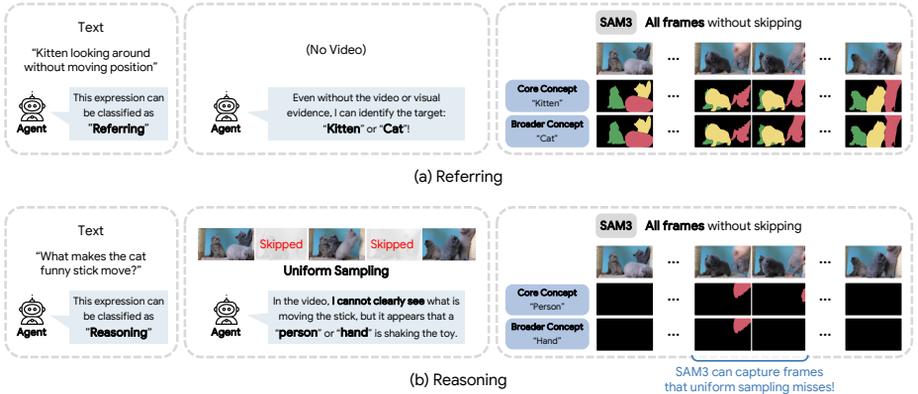
## A Additional Results and Analyses

### A.1 Ablation on the Effect of MLLM reasoning

**Table A1: Ablation on MLLM reasoning.** Replacing our MLLM-based reasoning pipeline with direct input of the referring expression into SAM3 leads to a substantial drop in segmentation accuracy and a sharp increase in empty mask ratio, demonstrating that MLLM reasoning is essential for handling the complexity of referring expressions.

Methods	Empty Mask Ratio (%)	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$
SAM3 [3]	38.9	36.8	44.7	40.8
<b>AgentRVOS</b>	3.8 (- 35.1)	67.1 (+ 30.3)	73.6 (+ 28.9)	70.3 (+ 29.5)

We evaluate the necessity of MLLM-based reasoning in our agentic pipeline by replacing it with direct input of the referring expression into SAM3 [3]. Since SAM3 is designed to process simple noun phrases rather than complex linguistic queries, this baseline bypasses query-grounded reasoning entirely. As shown in Tab. **A1**, removing MLLM reasoning leads to a substantial performance drop of 29.5 points in  $\mathcal{J}\&\mathcal{F}$  (40.8 vs. 70.3), alongside a sharp increase in the empty mask ratio from 3.8% to 38.9%, confirming that SAM3 alone cannot adequately handle the semantic complexity of referring expressions. These results demonstrate that the complementary roles of SAM3 and the MLLM are both necessary, where SAM3 provides reliable perceptual grounding while MLLM reasoning is essential for interpreting the query and resolving ambiguity among candidate tracks.

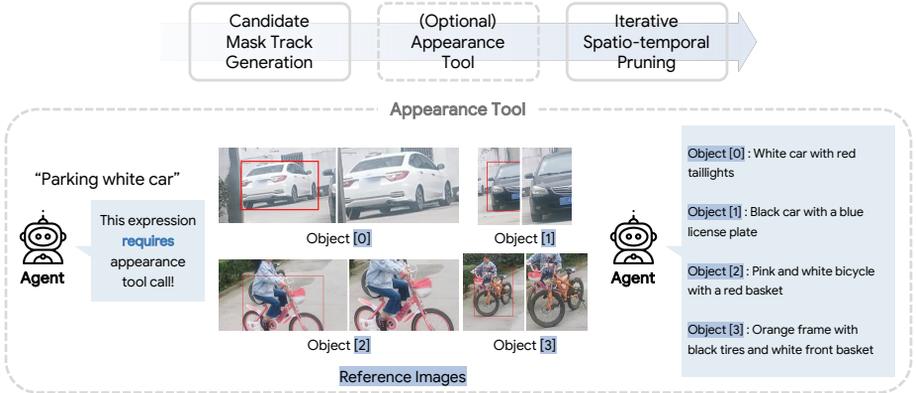


**Fig. A1: Concept extraction for referring and reasoning queries.** For *referring* queries (a), the target is identifiable from the expression alone, so SAM3 is applied directly using the extracted concept. For *reasoning* queries (b), resolving the referent requires visual evidence, so the video is provided alongside the expression. Notably, SAM3 is applied across all frames rather than a sampled subset, enabling reliable detection even when the target object appears in only a small fraction of the video, as illustrated by the “hand” visible in just two frames.

## A.2 Robustness of Concept Extraction

We provide further detail on the concept extraction step introduced in Sec. 3, focusing on how it handles diverse query types and why it reliably satisfies the coverage requirement of the candidate generation phase. The behavior of concept extraction is driven by the type of the given language query  $Q$ . Although this step may appear fragile—particularly for queries that resist direct noun extraction (e.g. “*what made the dogs move*”)—our pipeline explicitly accounts for the query type before invoking SAM3.

Specifically, we distinguish between two cases. For *referring* queries, the target object is identifiable from  $Q$  alone without visual evidence (Fig. A1 (a)). In this case, concept extraction proceeds from text only, and the resulting concepts reliably anchor SAM3 to the correct object category. For *reasoning* queries, the target object cannot be determined without observing the video (Fig. A1 (b)). Here, the MLLM examines uniformly sampled frames alongside  $Q$  to infer plausible candidate concepts. In most cases, this is sufficient to identify the target object. However, in extreme cases the referred object may not be clearly visible in the sampled subset—a hand appearing in only two frames, for instance, could be entirely absent from a uniform sample. Even so, the MLLM can still propose plausible candidates such as “person” or “hand” by reasoning about the expression and the available visual context. Because SAM3 processes all frames rather than only the sampled subset, it can verify whether these candidates actually exist anywhere in the video, reliably capturing objects that uniform sampling alone would miss.



**Fig. A2: Details of Appearance Tool.** Between candidate mask track generation and iterative spatio-temporal pruning, the appearance tool can be optionally invoked to obtain appearance information that may be obscured by visual prompting. When the agent determines that appearance-level evidence is needed for reasoning, it generates a brief phrase describing each candidate object’s appearance.

This design directly addresses the coverage requirement identified in Sec. 3: candidate tracks that do not contain the referred object cannot be recovered by subsequent reasoning. By converting the language query into SAM3-compatible noun phrases according to the query type, concept extraction ensures that the candidate pool is sufficiently complete before the spatio-temporal pruning phase begins.

### A.3 Details of Visual Prompting

In the iterative spatio-temporal pruning stage, the MLLM reasons over mask-overlaid videos where each candidate mask track is visualized with a distinct color [3, 39]. While this visual prompting strategy is essential for grounding the MLLM’s focus to specific object regions, it inevitably obscures the original appearance of each instance, including color, texture, and other fine-grained visual details. This can be problematic when the referring expression hinges on such appearance cues (*e.g.*, “*Parking white car*”).

To mitigate this, we introduce a simple appearance tool that can be optionally invoked before pruning begins. A brief illustration is provided in Fig. A2. When the MLLM determines that appearance-level evidence is needed, the tool constructs a two-panel image (which we refer to as a *reference image*) for each candidate following a similar visualization strategy to SAM3 Agent [3]: a loosely cropped view with the bounding box for spatial context, and a tightly cropped view for fine-grained appearance. Notably, for each candidate object, we select the single frame index where the object has the largest visible area, which is obtained from the mask track information produced by SAM3. The MLLM then generates a brief natural language description of each candidate’s appearance,

**Table A2: Ablation for appearance tool call.**

Component	$\mathcal{J}$	$\mathcal{F}$	$\mathcal{J}\&\mathcal{F}$
w/o Appearance Tool	64.6	70.9	67.7
<b>AgentRVOS</b>	67.1	73.6	70.3

which is carried forward into the pruning stage as supplementary evidence. Additionally, as shown in Tab. A2 (the results were obtained under the same setting as the ablation studies), incorporating the appearance tool yields some improvement, confirming that it effectively recovers appearance information lost under the visual overlay.

#### A.4 Detailed Algorithm

We provide a detailed description of the AgentRVOS pipeline through three algorithms. Algorithm 1 (Candidate Mask Track Generation) generates candidate mask tracks by extracting concept pairs from the query using an MLLM and prompting SAM3 with these concepts. Algorithm 2 (Appearance Tool) introduces an appearance tool that determines whether additional appearance information is required by the query and extracts it when necessary. Algorithm 3 (Iterative Spatio-temporal Pruning) performs iterative spatio-temporal pruning, where candidate mask tracks are progressively verified and filtered through reasoning over the video and the query.

**Candidate Mask Track Generation.** Algorithm 1 describes the procedure for generating mask tracks from a natural language query. Given a video  $\mathcal{V} = \{I_t\}_{t=1}^T$  and a language query  $Q$ , the algorithm iteratively extracts concept pairs using an MLLM and uses them to prompt SAM3 for mask track generation.

At the first iteration ( $k = 0$ ), the MLLM receives  $Q$  with  $\text{prompt}_{referring}$  (§ A9). The model first determines the query type. If the query corresponds to a *referring* case, where the referred object can be identified from the textual expression alone, the model directly extracts a set of concept pairs from the query. Otherwise, if the query is classified as *reasoning* and requires visual information from the video, the model does not extract concept pairs at this stage and instead proceeds to perform video conditioned concept extraction using  $\text{prompt}_{reasoning}$  (§ A10). Each extracted concept pair, indexed by  $i$ , consists of a core concept  $c_i^{core}$  that closely corresponds to the referred object and a broader concept  $c_i^{broad}$  that represents a more general category, increasing the likelihood of retrieving candidate tracks. For each concept pair  $(c_i^{core}, c_i^{broad})$ , SAM3 is applied to the video separately using the core concept and the broader concept, producing two candidate mask tracks. Among the concepts, we retain the track with the larger number of the sum of the total detected instances across frames, and denote this number as  $\text{COUNT}(\cdot)$ . If no instances are detected, a retry step is performed. Previously failed concept pairs are accumulated in a failure set and

provided to the MLLM in the next iteration to prevent the model from generating similar concepts again. This process continues until a mask track containing at least one detected instance is obtained or the maximum retry limit  $k_{max}$ , which is set to 3, is reached. The result of this stage is a set of candidate mask tracks  $M$  together with the final selected concepts  $C^*$  which provides class-level information about the target object for the subsequent selection stage.

**Appearance Tool.** Since mask overlays used during candidate visualization may obscure appearance cues such as color, an additional appearance tool is introduced to explicitly extract appearance information when needed. Algorithm 2 introduces an appearance tool that extracts additional appearance information when required by the query (e.g. “a white dog with black dots”). The MLLM first analyzes the query using `promptappearance_requirement` (§ A11 (top)) to determine whether appearance attributes are necessary for identifying the target object. If appearance information is required, the MLLM extracts appearance cues using `promptappearance_retrieval` (§ A11 (bottom)). The MLLM then receives reference images  $I$  for all candidate objects as described in Sec. A.3, and extracts an appearance description  $\mathcal{A}$  for all candidate objects.

**Iterative Spatio-temporal Pruning.** Algorithm 3 performs the final selection of the referred object through iterative spatio-temporal pruning of candidate mask tracks. The procedure begins with the candidate mask tracks  $M$  obtained from Algorithm 1. The initial temporal scope  $\mathcal{T}^{(0)}$  is defined as the union of frame indices where at least one instance is detected in the candidate mask tracks.

To proceed with the selection process, the candidate mask tracks are first visualized on the video frames, producing a masked overlaid video  $V^*$ . At each iteration  $r$ , frames are uniformly sampled from the current temporal scope  $\mathcal{T}^{(r)}$ . Each candidate mask track  $m_i \in M^{(r)}$  is then evaluated by the MLLM using `promptselect` (§ A12), conditioned on the mask overlaid video  $V^*$ , the query  $Q$ , the concept for each object  $C^*$ , and the appearance information  $\mathcal{A}$ .

Based on this evaluation, each mask track is classified into one of three categories: **Accepted**, **Rejected**, or **Uncertain**. Tracks classified as **Accepted** are added to the final mask set  $\mathcal{M}$ , while tracks labeled **Rejected** are removed from further consideration. Tracks labeled **Uncertain** are retained for further evaluation in the next iteration.

As the pruning process progresses, the temporal scope is updated using the temporal spans associated with the uncertain tracks. Specifically, the next temporal set  $\mathcal{T}^{(r+1)}$  is defined as the union of the temporal intervals  $\mathcal{T}(m_i)$  of the uncertain tracks. At the same time, visualization is also restricted to these uncertain candidates, which further reduces the spatial region under consideration. Together, these updates progressively narrow both the temporal and spatial search space to regions where ambiguous candidates remain.

The iterative pruning continues until no uncertain tracks remain or the maximum iteration limit  $r_{max}$ , which is set to 3, is reached. The final output of this stage is the set of predicted masks  $\mathcal{M}$ .

---

**Algorithm 1** Candidate Mask Track Generation
 

---

**Require:** Video  $\mathcal{V} = \{I_t\}_{t=1}^T$ , Language Query  $Q$ , MLLM  $\mathcal{L}$ , SAM3  $\mathcal{S}$   
**Ensure:** Candidate Mask Tracks  $M \in \{0, 1\}^{T \times H \times W}$ , Final selected concepts  $C^*$

- 1:  $k \leftarrow 0$  ▷ Iteration
- 2:  $M \leftarrow \emptyset$  ▷ Mask tracks
- 3:  $Fail \leftarrow \emptyset$  ▷ Failed Concept pairs
- 4:  $C^* \leftarrow \emptyset$
- 5: **while**  $M = \emptyset \wedge k \leq k_{max}$  **do**
- 6:   **if**  $k = 0$  **then**
- 7:      $A^{(k)} \leftarrow \mathcal{L}(Q, \text{prompt}_{referring})$  ▷ MLLM response
- 8:     **if**  $A^{(k)}[\text{query type}] = referring$  **then**
- 9:        $C^{(k)} \leftarrow A^{(k)}[\text{concept pairs}]$
- 10:     **else**
- 11:        $A^{(k)} \leftarrow \mathcal{L}(\mathcal{V}, Q, \text{prompt}_{reasoning})$  ▷ Video conditioned extraction
- 12:        $C^{(k)} \leftarrow A^{(k)}[\text{concept pairs}]$
- 13:     **end if**
- 14:     **else**
- 15:        $A^{(k)} \leftarrow \mathcal{L}(\mathcal{V}, Q, \text{prompt}_{reasoning}, Fail)$  ▷ Retry with failed concepts
- 16:        $C^{(k)} \leftarrow A^{(k)}[\text{concept pairs}]$
- 17:     **end if**
- 18:     **for each**  $(c_i^{core}, c_i^{broad}) \in C^{(k)}$  **do**
- 19:        $M_i^{core} \leftarrow \mathcal{S}(\mathcal{V}, c_i^{core})$
- 20:        $M_i^{broad} \leftarrow \mathcal{S}(\mathcal{V}, c_i^{broad})$
- 21:        $c_i^{selected} \leftarrow \text{argmax}_{c \in \{c_i^{core}, c_i^{broad}\}} \text{COUNT}(M_i^c)$  ▷ Concept selection
- 22:        $M_i^{selected} \leftarrow \mathcal{S}(\mathcal{V}, c_i^{selected})$
- 23:        $C^* \leftarrow C^* \cup c_i^{selected}$
- 24:        $M \leftarrow M \cup M_i^{selected}$
- 25:     **end for**
- 26:     **if**  $M = \emptyset$  **then**
- 27:        $Fail \leftarrow Fail \cup C^{(k)}$
- 28:     **end if**
- 29:      $k \leftarrow k + 1$
- 30: **end while**

---

## A.5 Visualization of Overall Reasoning Process

We provide end-to-end qualitative visualizations of AgentRVOS in Figs. A3 and A4, illustrating how the pipeline processes a referring expression from concept extraction through iterative spatio-temporal pruning to the final mask track output.

---

**Algorithm 2** Appearance Tool
 

---

**Require:** Language Query  $Q$ , MLLM  $\mathcal{L}$ , Reference images  $I$ 
**Ensure:** Appearance information  $\mathcal{A}$ 

- 1:  $B \leftarrow \mathcal{L}(Q, \text{prompt}_{\text{appearance\_requirement}})$  ▷ Check appearance requirement
  - 2: **if**  $B$  **then**
  - 3:    $\mathcal{A} \leftarrow \mathcal{L}(I, Q, \text{prompt}_{\text{appearance\_retrieval}})$  ▷ Extract appearance descriptions
  - 4: **else**
  - 5:    $\mathcal{A} \leftarrow \emptyset$
  - 6: **end if**
- 

---

**Algorithm 3** Iterative Spatio-Temporal Pruning
 

---

**Require:** Video  $\mathcal{V} = \{I_t\}_{t=1}^T$ , Language Query  $Q$ , MLLM  $\mathcal{L}$ , Candidate Mask Tracks  $M \in \{0, 1\}^{T \times H \times W}$ , Appearance information  $\mathcal{A}$ , Selected Concepts  $C^*$ 
**Ensure:** Predicted masks  $\mathcal{M} \in \{0, 1\}^{T \times H \times W}$ 

- 1:  $M^{(0)} \leftarrow M, \quad \mathcal{T}^{(0)} \leftarrow \bigcup_{m_i \in M^{(0)}} \mathcal{T}(m_i), \quad \mathcal{M} \leftarrow \emptyset$
  - 2: **for**  $r = 0, 1, \dots, r_{\max}$  **do**
  - 3:   Uniform sampled frames from  $\mathcal{T}^{(r)}$
  - 4:    $V^* \leftarrow \text{visualize}(\mathcal{V}, M^{(r)})$  ▷ Visualize mask tracks
  - 5:   Classify each  $m_i \in M^{(r)}$  as **Acc.**, **Rej.**, or **Unc.** via  $\mathcal{L}(V^*, Q, \text{prompt}_{\text{select}}, C^*, \mathcal{A})$
  - 6:    $\mathcal{M} \leftarrow \mathcal{M} \cup \{m_i \mid \text{Accepted}\}$
  - 7:    $M^{(r+1)} \leftarrow \{m_i \in M^{(r)} \mid \text{Uncertain}\}$
  - 8:    $\mathcal{T}^{(r+1)} \leftarrow \bigcup_{m_i \in M^{(r+1)}} \mathcal{T}(m_i)$  ▷ Narrow temporal scope
  - 9:   **if**  $M^{(r+1)} = \emptyset$  **then break**
  - 10:   **end if**
  - 11: **end for**
  - 12: **return**  $\mathcal{M}$
- 

In Fig. A3, the referring expression is “Which car disappears from the scene first?” The MLLM extracts the core concept “car” and the broader concept “vehicle.” Since SAM3 detects only 2 objects with the core concept, the pipeline falls back to the broader concept, yielding 4 candidate mask tracks. During iterative spatio-temporal pruning, the MLLM first rejects objects 2 and 3 as bicycles while classifying objects 0 and 1 as uncertain, since both cars remain visible in the sampled frames. In the subsequent iteration, with only the two cars remaining, the MLLM examines their temporal presence more carefully and determines that object 1 disappears first, accepting it as the final output. This example demonstrates a case where candidate pruning alone suffices to resolve the expression.

Fig. A4 presents a more complex case with the expression “Which creature has the minimum energy loss?” Again, the core concept “monkey” retrieves only 2 objects, so the pipeline falls back to “animal,” yielding 4 candidates. The first pruning iteration rejects the clearly active zebras (objects 2 and 3) while marking objects 4 and 5 as uncertain. In the next iteration, the MLLM observes that object 5 moves more noticeably and rejects it, but remains uncertain about object 4. At this point, temporal scope pruning is triggered: since object 4 mostly

does not span the full temporal range of the video, the pipeline restricts reasoning to the temporal scope where only object 4 exists. With this pruned context, the MLLM confirms that object 4 moves less actively compared to the zebras and accepts it as the final answer. This example shows how temporal scope pruning complements candidate pruning when spatial reasoning alone cannot fully resolve the expression.

## A.6 Additional Qualitative Results

We present additional qualitative results of AgentRVOS on MeViS [6] (Figs. A5, A6), ReVOS [38] (Fig. A7), and ReasonVOS [2] (Fig. A8). These examples span a diverse range of expression types, from motion-based descriptions and spatial relationships in MeViS, to causal and comparative reasoning in ReVOS, to commonsense and hypothetical reasoning in ReasonVOS. Across all benchmarks, AgentRVOS consistently produces accurate mask tracks, demonstrating its effectiveness.

## B Additional Implementation Details

### B.1 Ablation study configurations

All experiments are conducted via MLLM served through vLLM [18], with temperature set to 0.2 and a maximum of 8192 output tokens. Ablation experiments use Qwen3-VL-8B-Instruct [1] as the MLLM backbone and are evaluated on the MeViS *valid u* set.

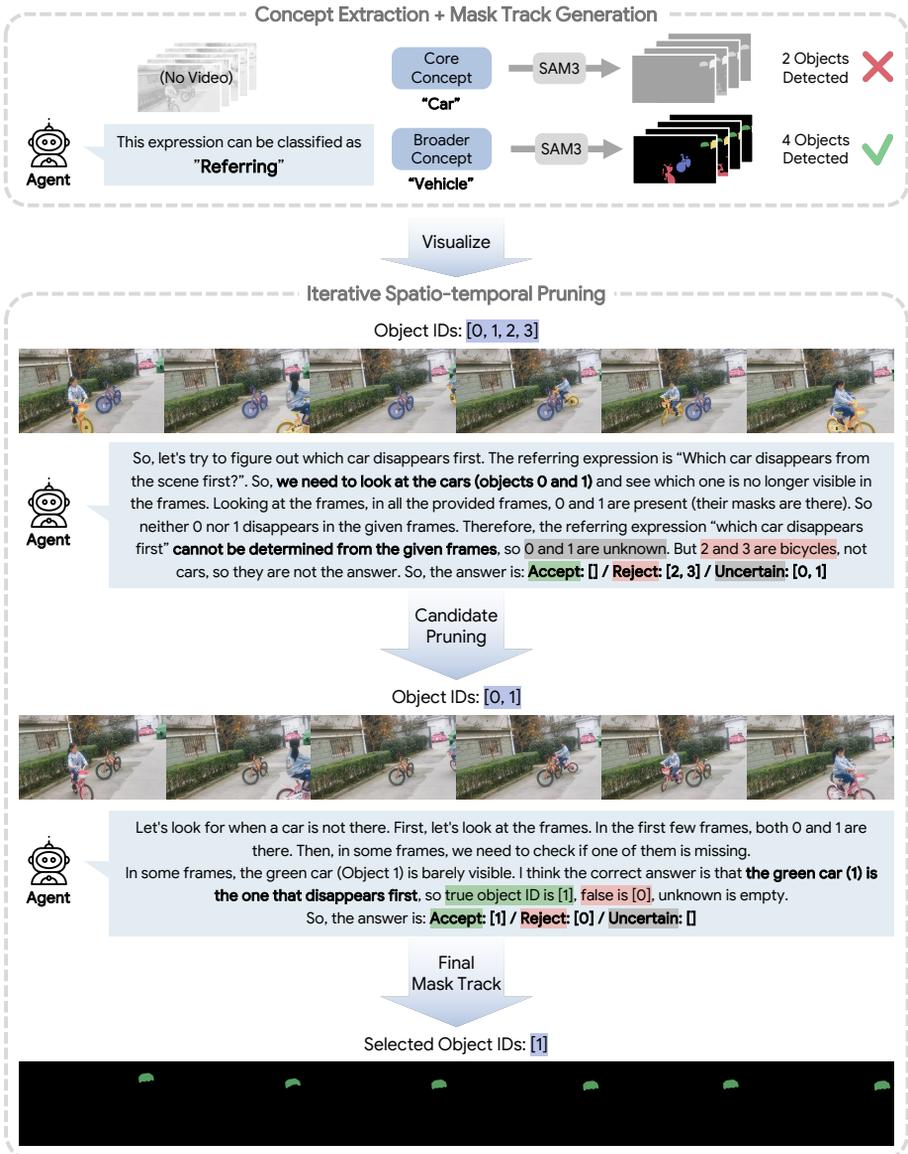
### B.2 Detailed prompts

We provide the prompts used in the AgentRVOS pipeline in Figs. A9, A10 (Concept Extraction), Fig. A11 (Appearance Tool), and Fig. A12 (Iterative Spatio-temporal Pruning).

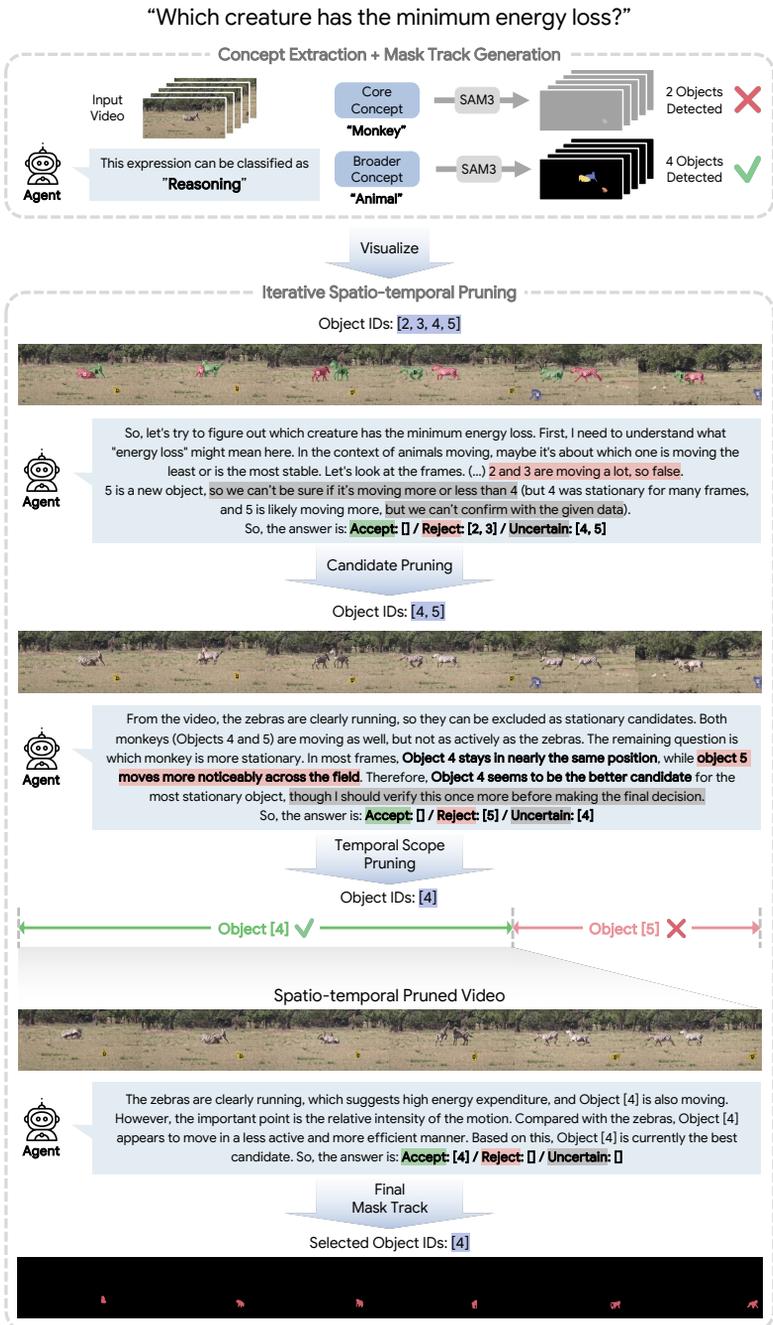
## C Future Works

As stronger MLLMs such as Gemini-3-Pro [31] continue to emerge, a promising direction is to substitute them into the pipeline, which requires no architectural modification due to the training-free and modular nature of AgentRVOS. To fully leverage their improved reasoning capabilities, exploring more sophisticated prompting strategies such as few-shot examples or structured chain-of-thought is another promising avenue for future work.

“Which car disappears from the scene first?”



**Fig. A3: Visualization of the overall pipeline of AgentRVOS.** This example illustrates a case where **candidate pruning** is applied. AgentRVOS progressively rejects irrelevant objects across iterations and identifies the correct target through proper iterative reasoning.



**Fig. A4: Visualization of the overall pipeline of AgentRVOS.** This example illustrates a case requiring both **candidate pruning** and **temporal scope pruning**. After spatially narrowing the candidates, the pipeline further restricts the temporal scope to resolve the remaining ambiguity.



“The distant parked motorcycle.”



“the little walks back and away”



“The rabbit that started off in the middle and later moved to the left”

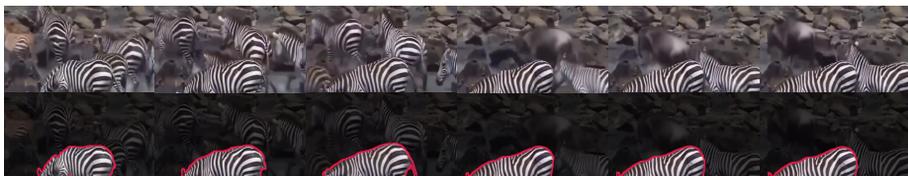


“The white pigeon that has moved a little from the left to the right.”

**Fig. A5: Qualitative results in MeViS.** MeViS [6] expressions require understanding object motion and spatial relationships. AgentRVOS successfully identifies targets based on motion descriptions such as walking direction and relative displacement.



“The pigeon that walked a few steps to the left.”



“The zebra remaining stationary without any movement.”



“A monkey squatting in the distant right without moving.”



“The elephant that drank water first.”

**Fig. A6: Qualitative results in MeViS.** These examples involve distinguishing among multiple similar objects by their motion patterns, such as remaining stationary or performing a specific action first.



"The vehicle that overtakes from the left and heads in a different direction at the intersection."



"When your hands are wet from washing, which tool can help soak up the excess water?"



"The object that was temporarily in the girl's left hand grasp."



"In case of a flat tire, which vehicle would have a spare tire on board?"

**Fig. A7: Qualitative results in ReVOS.** ReVOS [38] expressions often involve complex reasoning about object interactions and scene context. AgentRVOS correctly segments targets described through causal relationships, relative comparisons, and challenging visual conditions.



"Which object(s) was/were put into the dustpan and finally fell into the pit?"



"Which object has the largest displacement?"



"the dog that is not easily found in the dark."



"Which object(s) might be picked up by the man in black?"

**Fig. A8: Qualitative results in ReasonVOS.** ReasonVOS [2] requires common-sense and world knowledge beyond direct visual cues. AgentRVOS handles expressions involving functional reasoning, hypothetical scenarios, and temporal event understanding.

**Candidate Mask Track Generation – Referring**

You are a linguistic specialist for Referring Video Object Segmentation (RVOS).

Your Primary Objective:  
Analyze the user's expression and determine if the **Target Object Class** can be identified explicitly from the text alone.

(...)

**### Classification Logic:**

- 1. EXPLICIT (Specific Class Known)**
  - \* The identified target is a **specific physical noun** (e.g., "dog", "car", "plate", "man").
  - \* Even with typos ("yhe") or broken grammar ("bear fight"), if the specific class is clear, output EXPLICIT.
- 2. REASONING (Visual Context Required)**
  - \*\*Questions / Interrogatives:\*\*** The expression is a question asking about the identity of the target.
  - \*\*Generic Placeholders:\*\*** The target is identified grammatically, but the noun itself is too vague to determine the class without video.
  - \* **Logic:** In "The **object** that is scared by the monkey", the target is **object**. Since "object" is generic, you must look at the video to know what it is. -> Output **REASONING**.\*
  - \* **Keywords:** "object", "thing", "item", "stuff", "entity", "one".
  - \*\*Non-Physical / Intangible:\*\*** The identified target is an **abstract concept, phenomenon, or intangible entity**, NOT a physical object class.
  - \* **Keywords:** "sound", "voice", "speed", "shadow", "reflection", "air", "time", "moment".
  - \*\*Ambiguous:\*\*** The target is a pronoun (it, they) or interrogative (what, who).

**### Output Format (JSON Only):**

```
{
  "expression_type": "EXPLICIT" or "REASONING",
  "nouns": [{"base_noun", "hypernym"}] or null,
  "reasoning": "Brief explanation. Format: 'Target is [Noun] ([Reason)].'"
}
```

\* **base\_noun:** Singular form (e.g., "monkeys" -> "monkey").

\* **SAM-Optimized Hypernym Mapping (CRITICAL)**  
Map the extracted noun to a **Common Visual Superclass** optimized for AI segmentation models (like SAM).

**Rule A: Rare Species -> Visual Proxy (Shape Match)**

- \* If the noun is a rare species or specific type, map it to the **most common visual equivalent** that shares the same shape.
- \* Examples: "Yak" -> **Cow**, "Baboon" -> **Monkey**.

**Rule B: Common Objects -> Broad Category**

- \* If the noun is a common object, map it to its standard superclass.
- \* Examples: "Car" -> **Vehicle**, "Truck" -> **Vehicle**.

\* **Identity Rule:** If the base noun is already a top-level generic category, set to `null`.

\* **Reasoning:** concise justification focusing **ONLY** on the identified target and its specificity. Do NOT list ignored context words.

**Fig. A9: Prompts used in AgentRVOS (prompt<sub>referring</sub>).** The prompt for concept extraction, where the model receives a referring expression and extracts a core concept and a broader concept to guide SAM3 mask track generation.

## Candidate Mask Track Generation – Reasoning

You are a Video Reasoning Expert for Referring Video Object Segmentation (RVOS).

Your task is to identify the specific **Physical Object(s)** corresponding to the provided text expression by analyzing the video frames.

### Input Parameters:

1. **Expression Type:** 'EXPLICIT' (Text names the object) or 'REASONING' (Text is ambiguous/question).
2. **Iteration:** Current attempt number (1, 2, ...).
3. **Failed Nouns:** List of nouns that failed in previous attempts. **\*\*DO NOT output these again.\*\***

### Workflow by Expression Type:

#### CASE A: Expression Type == 'REASONING' (Ambiguous / Questions)

**\*\*Goal:\*\*** Visually infer the missing target.

**\* Logic:**

**Step 1: Analyze Ambiguity Type**

Identify why the text was insufficient and what to look for in the video:

- \*\*Pronouns ("It", "Ones"):\*\*** Find the specific object the pronoun refers to.
- \*\*Interrogatives ("Who", "What"):\*\*** Visually answer the question.
- \*\*Abstract/Intangible ("Pressure", "Force", "Wind"):\*\*** Find the **\*\*Visible Source/Agent\*\*** causing the effect.
  - \*Example:\* "Invisible pressure making the dog move" -> Look for **\*what\*** scares the dog (e.g., a vacuum cleaner).
- \*\*Missing Subject ("Moving fast"):\*\*** Find the actor performing the action.

**Step 2: Visual Inference & Extraction (Best Effort)**

- \*\*Visible:\*\*** If the target object is clearly visible in the frames, extract its **\*\*Singular Base Noun\*\*** directly.
  - \*Rule:\* Convert plural to singular (e.g., "cars" -> **\*\*car\*\***, "children" -> **\*\*child\*\***).
- \*\*Not Visible / Ambiguous:\*\*** If the exact target is not visible or clear, **\*\*infer the most plausible visible object\*\*** based on the visual context.
  - \*Strategy:\* Identify the object being looked at, the surface receiving an action, or the visible proxy causing the effect. Just make the best logical guess from what is shown and output the **\*\*Singular Form\*\***.

#### CASE B: Expression Type == 'EXPLICIT' (Specific Naming)

**\*\*Goal:\*\*** Translate the specific name into a **\*\*SAM-Friendly Visual Descriptor\*\***.

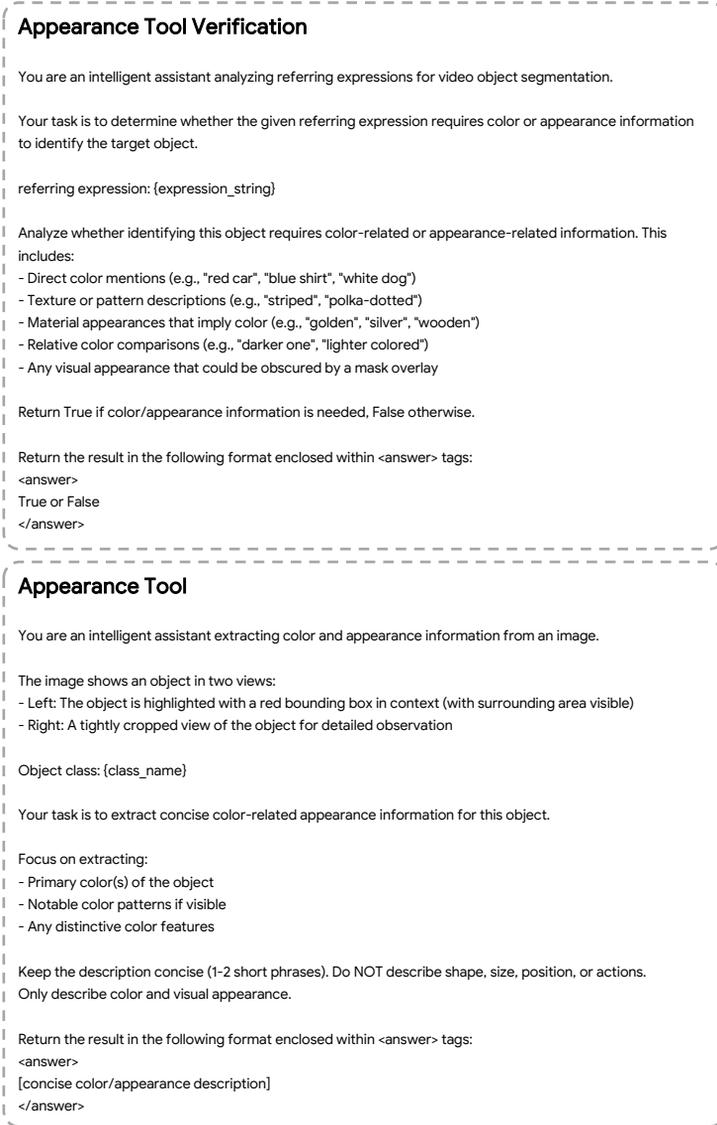
**\*\*Logic:\*\***

1. **\*\*Avoid Specifics:\*\*** SAM models struggle with proper nouns (e.g., "Mario", "Pororo") or highly specific sub-types.
2. **\*\*Visual Translation:\*\*** Convert the specific name into a description of its **\*\*visual appearance\*\*** or **\*\*material\*\***.
3. **\*\*Generalization:\*\*** If **'iteration' > 1**, broaden the category significantly.

### Output Format (JSON Only):

```
{
  "nouns": [{"base_noun": "hypernym"}],
  "reasoning_logic": "Brief text explaining the visual inference (e.g., 'Text says 'pressure', video shows a vacuum cleaner scaring the dog, so the target is the vacuum cleaner.')"
}
```

**Fig. A10: Prompts used in AgentRVOS (prompt<sub>reasoning</sub>).** The prompt for concept extraction, where the model receives the referring expression and extracts a core concept and a broader concept to guide SAM3 mask track generation.



**Fig. A11: Prompts used in AgentRVOS** ( $\text{prompt}_{\text{appearance\_requirement}}$  and  $\text{prompt}_{\text{appearance\_retrieval}}$ ). The prompt for Appearance Tool Verification (top) determines whether the referring expression requires appearance-level evidence, and the prompt for Appearance Tool (bottom) extracts concise color and appearance descriptions from each candidate.

## Iterative Spatio-temporal Pruning

You are an intelligent assistant for generalized referring expression segmentation and reasoning segmentation.

User has provided a referring expression and uniformly sampled frames from video.

The frames are visually prompted with object masks and object IDs.

Your task is to partition object IDs into true, false, and unknown object IDs according to referring expression.

referring expression: {expression\_string}

object IDs: {unknown\_obj\_ids}

Please analyze the visually prompted frames and partition the object IDs into:

1. `true_obj_ids`: The object IDs that definitely DO match the referring expression with high confidence.
2. `false_obj_ids`: The object IDs that definitely DO NOT match the referring expression with high confidence.
3. `unknown_obj_ids`: The object IDs for which it cannot be clearly determined whether they match the referring expression.

IMPORTANT: Your response MUST include ALL object IDs from the input list: {unknown\_obj\_ids}

Each ID must appear in exactly ONE of the three lists. Missing any ID is an error.

Return the result in the following JSON format enclosed within `<answer>` tags:

```
<answer>
{{
  "true_obj_ids": [list of true object IDs],
  "false_obj_ids": [list of false object IDs],
  "unknown_obj_ids": [list of unknown object IDs]
}}
```

**Fig. A12: Prompts used in AgentRVOS (prompt<sub>select</sub>).** The prompt for Iterative Spatio-temporal Pruning, where the model receives a candidate mask track and a language query and classifies each candidate as **Accept**, **Reject**, or **Uncertain**, progressively narrowing the candidate pool.

## References

1. Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., et al.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025) [4](#), [9](#), [22](#)
2. Bai, Z., He, T., Mei, H., Wang, P., Gao, Z., Chen, J., Liu, L., Zhang, Z., Shou, M.Z.: One token to seg them all: Language instructed reasoning segmentation in videos. arXiv preprint arXiv:2409.19603 (2024). <https://doi.org/10.48550/arXiv.2409.19603>, <https://arxiv.org/abs/2409.19603> [2](#), [4](#), [8](#), [9](#), [22](#), [28](#)
3. Carion, N., Hu, Y.T., Lin, H., Krähenbühl, P., Yu, Q., Majewski, K., Nimier-David, N., et al.: Sam 3: Segment anything with concepts. arXiv preprint arXiv:2511.16719 (2025). <https://doi.org/10.48550/arXiv.2511.16719>, <https://arxiv.org/abs/2511.16719> [3](#), [5](#), [7](#), [9](#), [15](#), [17](#)
4. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European conference on computer vision. pp. 213–229. Springer (2020) [4](#)
5. Chen, Z., Wang, W., Cao, Y., Liu, Y., Gao, Z., Cui, E., Zhu, J., Ye, S., Tian, H., Liu, Z., Gu, L., Wang, X., Li, Q., Ren, Y., Chen, Z., Luo, J., Wang, J., Jiang, T., Wang, B., He, C., Shi, B., Zhang, X., Lv, H., Wang, Y., Shao, W., Chu, P., Tu, Z., He, T., Wu, Z., Deng, H., Ge, J., Chen, K., Zhang, K., Wang, L., Dou, M., Lu, L., Zhu, X., Lu, T., Lin, D., Qiao, Y., Dai, J., Wang, W.: Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling (2025), <https://arxiv.org/abs/2412.05271> [4](#)
6. Ding, H., Liu, C., He, S., Jiang, X., Loy, C.C.: Mevis: A large-scale benchmark for video segmentation with motion expressions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 2694–2703 (October 2023), [https://openaccess.thecvf.com/content/ICCV2023/papers/Ding\\_MeViS\\_A\\_Large-scale\\_Benchmark\\_for\\_Video\\_Segmentation\\_with\\_Motion\\_Expressions\\_ICCV\\_2023\\_paper.pdf](https://openaccess.thecvf.com/content/ICCV2023/papers/Ding_MeViS_A_Large-scale_Benchmark_for_Video_Segmentation_with_Motion_Expressions_ICCV_2023_paper.pdf) [1](#), [4](#), [8](#), [9](#), [22](#), [25](#)
7. Gavriluyk, K., Ghodrati, A., Li, Z., Snoek, C.G.M.: Actor and action video segmentation from a sentence. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018), [https://openaccess.thecvf.com/content\\_cvpr\\_2018/papers/Gavriluyk\\_Actor\\_and\\_Action\\_CVPR\\_2018\\_paper.pdf](https://openaccess.thecvf.com/content_cvpr_2018/papers/Gavriluyk_Actor_and_Action_CVPR_2018_paper.pdf) [4](#)
8. Gong, S., Zhuge, Y., Zhang, L., Yang, Z., Zhang, P., Lu, H.: The devil is in temporal token: High quality video reasoning segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 29183–29192 (2025) [9](#)
9. He, S., Ding, H.: Decoupling static and hierarchical motion perception for referring video segmentation (2024), <https://arxiv.org/abs/2404.03645> [4](#)
10. Huang, S., Ling, R., Li, H., Hui, T., Tang, Z., Wei, X., Han, J., Liu, S.: Al-ref-sam 2: Unleashing the temporal-spatial reasoning capacity of gpt for training-free audio and language referenced video object segmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 3715–3723 (2025). <https://doi.org/10.48550/arXiv.2408.15876>, <https://arxiv.org/abs/2408.15876> [2](#), [9](#)
11. Hurst, A., Lerer, A., Goucher, A.P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A., Welihinda, A., Hayes, A., Radford, A., et al.: Gpt-4o system card. arXiv preprint arXiv:2410.21276 (2024) [2](#), [9](#)

12. Jiang, H., Liang, T., Zheng, W.S., Hu, J.F.: Refer-agent: A collaborative multi-agent system with reasoning and reflection for referring video object segmentation. arXiv preprint arXiv:2602.03595 (2026) **2, 4**
13. Jin, W., Kim, S., Lee, J., Kim, S.: Intervros: Interaction-aware referring video object segmentation. arXiv preprint arXiv:2506.02356 (2025) **1**
14. hong Kao, S., Huang, C.H., Liu, H., Tai, Y.W., Tang, C.K.: Cot-seg: Rethinking segmentation with chain-of-thought reasoning and self-correction (2026), <https://arxiv.org/abs/2601.17420> **1**
15. Kao, S.h., Tai, Y.W., Tang, C.K.: Cot-rvs: Zero-shot chain-of-thought reasoning segmentation for videos. arXiv preprint arXiv:2505.18561 (2025) **2, 4, 8, 9, 13**
16. Khoreva, A., Rohrbach, A., Schiele, B.: Video object segmentation with referring expressions. In: The European Conference on Computer Vision (ECCV) Workshops (September 2018), [https://openaccess.thecvf.com/content\\_ECCVW\\_2018/papers/11132/Khoreva\\_Video\\_Object\\_Segmentation\\_with\\_Referring\\_Expressions\\_ECCVW\\_2018\\_paper.pdf](https://openaccess.thecvf.com/content_ECCVW_2018/papers/11132/Khoreva_Video_Object_Segmentation_with_Referring_Expressions_ECCVW_2018_paper.pdf) **4**
17. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollár, P., Girshick, R.: Segment anything (2023), <https://arxiv.org/abs/2304.02643>
18. Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C.H., Gonzalez, J., Zhang, H., Stoica, I.: Efficient memory management for large language model serving with pagedattention. In: Proceedings of the 29th symposium on operating systems principles. pp. 611–626 (2023) **22**
19. Lai, H.H., et al.: Lisa: Reasoning segmentation via large language model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2024), [https://openaccess.thecvf.com/content/CVPR2024/papers/Lai\\_LISA\\_Reasoning\\_Segmentation\\_via\\_Large\\_Language\\_Model\\_CVPR\\_2024\\_paper.pdf](https://openaccess.thecvf.com/content/CVPR2024/papers/Lai_LISA_Reasoning_Segmentation_via_Large_Language_Model_CVPR_2024_paper.pdf) **1**
20. Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., et al.: Llava-onevision: Easy visual task transfer. arXiv preprint arXiv:2408.03326 (2024) **2, 4**
21. Li, Y., Yin, Y., Zhu, L., Chen, W., Qian, S., Wang, X., Fu, Y.: Revseg: Incentivizing the reasoning chain for video segmentation with reinforcement learning (2025), <https://arxiv.org/abs/2512.02835> **4**
22. Lin, L., Yu, X., Pang, Z., Wang, Y.X.: Glus: Global-local reasoning unified into a single large language model for video segmentation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 8658–8667 (2025) **2, 9**
23. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. *Advances in neural information processing systems* **36**, 34892–34916 (2023) **2**
24. Liu, Y., Peng, B., Zhong, Z., Yue, Z., Lu, F., Yu, B., Jia, J.: Seg-zero: Reasoning-chain guided segmentation via cognitive reinforcement (2025), <https://arxiv.org/abs/2503.06520> **1**
25. Miao, B., Bennamoun, M., Gao, Y., Shah, M., Mian, A.: Temporally consistent referring video object segmentation with hybrid memory. *IEEE Transactions on Circuits and Systems for Video Technology* (2024) **4**
26. OpenAI: Introducing gpt-5 (Aug 2025), august 7 2025 **9**
27. Qwen, :, Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., Lin, H., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Lin, J., Dang, K., Lu, K., Bao, K., Yang, K., Yu, L., Li, M., Xue, M., Zhang, P., Zhu, Q., Men, R., Lin, R., Li, T., Tang, T., Xia, T., Ren, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Wan, Y., Liu, Y., Cui, Z., Zhang, Z., Qiu, Z.: Qwen2.5 technical report (2025), <https://arxiv.org/abs/2412.15115> **4**

28. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PmLR (2021) [4](#)
29. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., Mintun, E., Pan, J., Alwala, K.V., Carion, N., Wu, C.Y., Girshick, R., Dollár, P., Feichtenhofer, C.: Sam 2: Segment anything in images and videos. arXiv preprint arXiv:2408.00714 (2024). <https://doi.org/10.48550/arXiv.2408.00714>, <https://arxiv.org/abs/2408.00714> [2](#), [4](#)
30. Seo, S., Lee, J.Y., Han, B.: Urvos: Unified referring video object segmentation network with a large-scale benchmark. In: European Conference on Computer Vision (ECCV). pp. 208–223 (2020). [https://doi.org/10.1007/978-3-030-58555-6\\_13](https://doi.org/10.1007/978-3-030-58555-6_13), [https://doi.org/10.1007/978-3-030-58555-6\\_13](https://doi.org/10.1007/978-3-030-58555-6_13) [4](#)
31. Team, G., Anil, R., Borgeaud, S., Alayrac, J.B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A.M., Hauth, A., Millican, K., et al.: Gemini: a family of highly capable multimodal models. arXiv preprint arXiv:2312.11805 (2023) [2](#), [22](#)
32. Varma, M., Delbrouck, J.B., Hooper, S., Chaudhari, A., Langlotz, C.: Villa: Fine-grained vision-language representation learning from real-world data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 22225–22235 (2023) [9](#)
33. Wang, H., Chen, Q., Yan, C., Cai, J., Jiang, X., Hu, Y., Xie, W., Gavves, S.: Object-centric video question answering with visual grounding and referring. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 22274–22284 (2025) [9](#)
34. Wei, C., Zhong, Y., Tan, H., Liu, Y., Zhao, Z., Hu, J., Yang, Y.: Hyperseg: Towards universal visual segmentation with large language model. arXiv preprint arXiv:2411.17606 (2024) [9](#)
35. Wei, C., Zhong, Y., Tan, H., Zeng, Y., Liu, Y., Wang, H., Yang, Y.: Instructseg: Unifying instructed visual segmentation with multi-modal large language models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 20193–20203 (2025) [9](#)
36. Wu, J., Jiang, Y., Sun, P., Yuan, Z., Luo, P.: Language as queries for referring video object segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4974–4984 (June 2022), [https://openaccess.thecvf.com/content/CVPR2022/papers/Wu\\_Language\\_As\\_Queries\\_for\\_Referring\\_Video\\_Object\\_Segmentation\\_CVPR\\_2022\\_paper.pdf](https://openaccess.thecvf.com/content/CVPR2022/papers/Wu_Language_As_Queries_for_Referring_Video_Object_Segmentation_CVPR_2022_paper.pdf) [4](#)
37. Xu, Q., Lian, L., Wang, Z., Long, Q., Gao, Y., Hu, W.: Videoseg-r1: Reasoning video object segmentation via reinforcement learning. arXiv preprint arXiv:2511.16077 (2025). <https://doi.org/10.48550/arXiv.2511.16077>, <https://arxiv.org/abs/2511.16077> [4](#), [9](#)
38. Yan, C., et al.: Visa: Reasoning video object segmentation via large language models. arXiv preprint arXiv:2407.11325 (2024). <https://doi.org/10.48550/arXiv.2407.11325>, <https://arxiv.org/abs/2407.11325> [1](#), [2](#), [4](#), [8](#), [9](#), [22](#), [27](#)
39. Yang, J., Zhang, H., Li, F., Zou, X., Li, C., Gao, J.: Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. arXiv preprint arXiv:2310.11441 (2023) [7](#), [17](#)
40. Yuan, J., Zhang, T., Sun, H., Yang, Y., Yuan, S., Loy, C.C.: Sa2va: Marrying sam2 with llava for dense grounded understanding of images and videos. arXiv preprint arXiv:2501.04001 (2025). <https://doi.org/10.48550/arXiv.2501.04001>, <https://arxiv.org/abs/2501.04001> [2](#), [4](#), [9](#)