

Can Vision-Language Models Solve the Shell Game?

Tiedong Liu

National University of Singapore
tiedong.liu@u.nus.edu

Wee Sun Lee

National University of Singapore
dcsleews@nus.edu.sg

Abstract

Visual entity tracking is an innate cognitive ability in humans, yet it remains a critical bottleneck for Vision-Language Models (VLMs). This deficit is often obscured by visual shortcuts in existing video benchmarks. We introduce VET-Bench, a synthetic diagnostic testbed featuring visually identical objects that necessitate tracking exclusively through spatiotemporal continuity. Our experiments reveal that current state-of-the-art VLMs perform at or near chance level on VET-Bench. We provide a theoretical analysis drawing connections to the state-tracking problem, suggesting that fixed-depth transformer-based VLMs are fundamentally limited in tracking indistinguishable objects without intermediate computation due to expressivity constraints. To address this, we propose Spatiotemporal Grounded Chain-of-Thought (SGCoT): generating object trajectories as explicit intermediate states. Leveraging Molmo2’s object-tracking capability, we elicit SGCoT reasoning by fine-tuning on synthesized text-only data. Our method achieves state-of-the-art accuracy exceeding 90% on VET-Bench, demonstrating that VLMs can reliably solve the video shell-game task end-to-end without external tools. Our code and data are available at <https://vetbench.github.io>.

1 Introduction

Vision-Language Models (VLMs) have demonstrated remarkable proficiency in general video understanding and reasoning (Wu et al., 2024; Wang et al., 2025; Fu et al., 2025; Hu et al., 2026). However, their low-level perception, specifically the ability to track entities over time, remains a critical bottleneck. We investigate how video VLMs perform on visual entity tracking tasks, such as the shell game. While such tasks are often effortless for humans and even some animals (Jaakkola, 2014), they present significant challenges for current VLMs, highlighting a key limitation in fine-

grained spatiotemporal perception, which is essential for many downstream applications, such as embodied AI (Thompson et al., 2025; Fung et al., 2025) and general game-playing agents (SIMA Team et al., 2025; Magne et al., 2026).

Existing video benchmarks such as the Perception Test (Patraucean et al., 2023) attempt to evaluate this capability using real-world recordings of the shell game (i.e., cups-game subset). However, our audit reveals that many cups-game clips contain appearance cues: distinct or transparent cups allow models to solve the task by re-identifying the object from a single frame, rather than performing full temporal tracking across frames. Our findings suggest that visual entity tracking is a key limiting factor for model performance on the Perception Test. Easier instances requiring little to no tracking can yield high accuracies, inflating the aggregate score and obscuring failures on genuinely tracking-dependent cases. Consequently, we argue that the shell-game task exposes a critical failure mode: addressing this task is an important step toward genuine, human-level visual perception.

To systematically investigate this limitation, we introduce the **Visual Entity Tracking Benchmark (VET-Bench)**, a synthetic diagnostic testbed designed to isolate spatiotemporal perception from frame-level appearance cues. By using sets of visually identical objects, VET-Bench forces models to track entities solely through motion continuity. Our extensive experiments reveal that current state-of-the-art models, such as Gemini 3 Pro (Pichai et al., 2025), perform at or near chance on VET-Bench. These results align with our audit of the Perception Test, suggesting that models rely heavily on static appearance features rather than genuine entity tracking. When these discriminative appearance cues are removed, VLMs struggle to maintain coherent entity representations over time.

Humans solve the shell game through an intuitive perceptual process that requires little con-

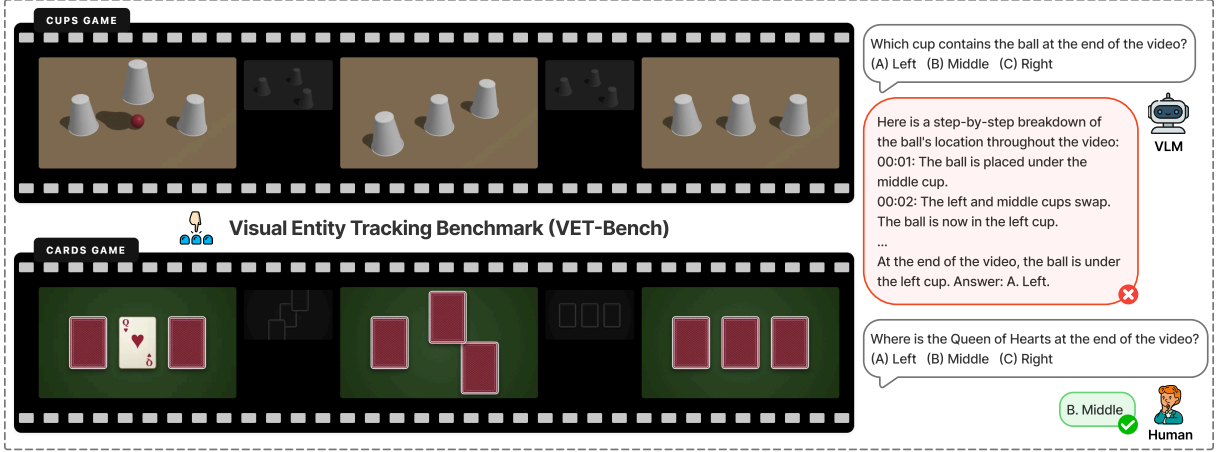


Figure 1: Overview of VET-Bench.

scious effort. This raises a fundamental question: are transformer-based (Vaswani et al., 2017) VLMs inherently capable of solving the video shell game? By drawing connections to the state-tracking problem, we provide a theoretical analysis proving that visual entity tracking is **NC¹-complete**. This suggests that fixed-depth transformers are fundamentally limited in solving general visual entity tracking tasks without intermediate computation due to expressivity constraints (Merrill and Sabharwal, 2023, 2024; Feng et al., 2023; Li et al., 2024d; Huang et al., 2025). We empirically verify that VLMs struggle to learn the shell game using direct-answer supervision.

To address this, we propose **Spatiotemporal Grounded Chain-of-Thought (SGCoT)**. Leveraging the object-tracking capability of Molmo2 (Clark et al., 2026), we transform perception into a reasoning process where the model explicitly generates spatiotemporally grounded object trajectories as intermediate states before providing the final answer. By fine-tuning on synthetic text-only data for alignment, we elicit SGCoT in Molmo2 and achieve state-of-the-art accuracy exceeding 90% on VET-Bench, demonstrating that VLMs can reliably solve the video shell-game task end-to-end without external tools.

Our work offers the following contributions:

- We identify visual entity tracking as a critical bottleneck in the Perception Test benchmark and introduce VET-Bench, a synthetic diagnostic benchmark featuring visually identical objects, providing a rigorous testbed for evaluating the visual perception of VLMs by eliminating appearance-based shortcuts.

- We conduct a comprehensive evaluation of state-of-the-art proprietary and open-source video VLMs, revealing that all tested models perform near chance on VET-Bench.
- We provide an NC¹-completeness proof for the visual entity tracking task, establishing the necessity of intermediate computation for transformer-based VLM architectures. We empirically verify that a VLM trained only on direct answers without intermediate computation fails to learn the shell game.
- We demonstrate that VLMs are capable of solving the shell game end-to-end without external tools through SGCoT. By transforming perception into a reasoning process and aligning Molmo2 to generate object trajectories as CoT, we achieve state-of-the-art performance exceeding 90% accuracy on VET-Bench.

2 VET-Bench

2.1 Task Formulation

We consider a video sequence $\mathcal{V} = (F_0, \dots, F_T)$ containing N visually indistinguishable objects, identified by their initial indices $i \in [N] = \{1, \dots, N\}$. The shuffling process induces a global permutation π , mapping the object at initial index i to its final index $\pi(i)$. Given a target object $i \in [N]$ designated at $t = 0$ via a contextual cue, which may take the form of a visual highlight or a linguistic description, the objective is to determine its terminal index $\pi(i)$ in the final frame.

To ensure the task is well-posed without ambiguity, we enforce a continuity constraint: the maximum displacement d of any object between

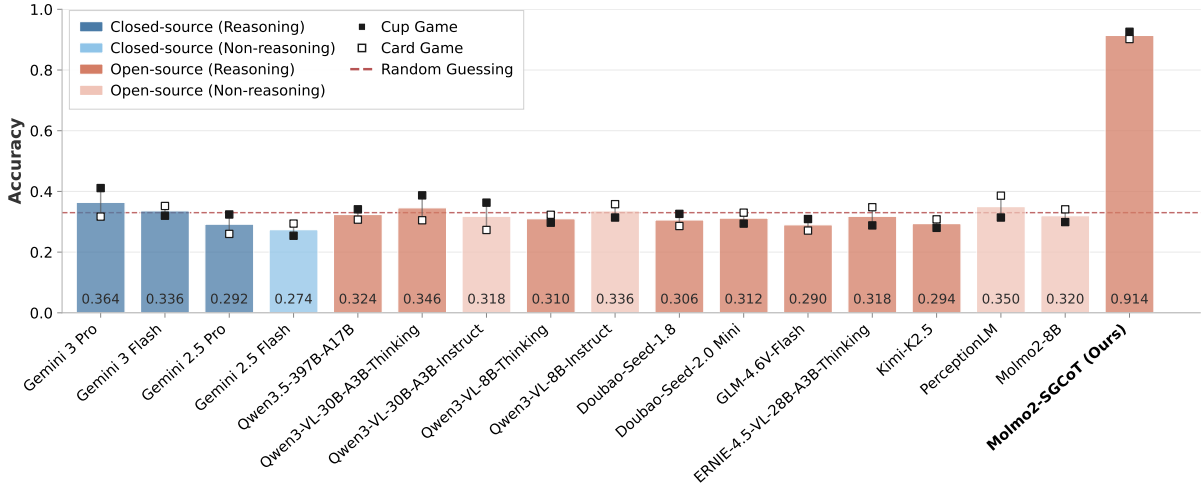


Figure 2: Performance on VET-Bench, consisting of 250 cups-game and 250 cards-game videos featuring 3 objects and 5 swaps (~ 12 seconds). Existing VLMs all perform near chance. Molmo2-SGCoT is a fine-tuned model based on Molmo2-8B that leverages Spatiotemporal Grounded Chain-of-Thought (Section 5).

consecutive frames must satisfy $2d < \Delta$, where Δ is the minimum spatial separation between any two objects. This constraint mirrors the temporal Nyquist criterion in video sampling and prevents identity aliasing during object crossovers. In real-world videos, this condition is naturally satisfied by a sufficiently high frame rate relative to object speeds. In VET-Bench, each swap lasts 2 seconds, so the minimum sampling rate required to resolve a swap without ambiguity is 1 FPS (Section 3.1).

2.2 Data Generation

We target the following two canonical visual entity tracking tasks (Fig. 1):

- **Cups Game.** Also known as the shell game, this task requires tracking a ball hidden beneath visually identical opaque containers that undergo positional swaps.
- **Cards Game.** Modeled after Three-Card Monte (“Find the Queen”), this task requires tracking a card after it is flipped face down and shuffled.

Inspired by the synthetic design of CLEVR (Johnson et al., 2017), CLEVRER (Yi et al., 2020), and CATER (Girdhar and Ramanan, 2020), our programmatic data generation pipeline renders videos via `three.js`, enabling full control over environmental attributes such as color, material, texture, lighting, and camera viewpoint. Precise control over parameters such as the number of objects and swaps enables targeted diagnostic evaluation (Section 3.3). This setup not only mitigates data leakage

and overfitting, but also allows us to generate an unlimited number of episodes to support model training.

2.3 Limitations of Existing Benchmarks

While some existing benchmarks include shell-game-style tasks, they are less diagnostic due to uncontrolled complexity or visual shortcuts.

The Perception Test. The Perception Test (Patraucean et al., 2023) includes a small subset of cups-game clips, where the number of cups (2–4) and swap counts (0–16) vary widely. As shown in Section 3.3, both variables affect model performance. Moreover, many clips offer visual shortcuts (Figs. 4 to 6)—such as distinct or transparent cups—that allow re-identification without tracking. We filtered the test split (3,525 videos) down to a standard shell-game setting: three identical, opaque cups and at least one swap. This filtering reduced 186 initial cups-game clips to 107 shortcut-free clips, and finally to 65 3-cup clips with at least one swap (Fig. 8). Model performance collapses on this subset (Table 1). For example, Gemini 3 Pro drops from 0.80 on the full dataset to 0.31, effectively random guessing. However, a filtered subset of 65 videos is too small to serve as a reliable benchmark due to high variance in configurations. Furthermore, these real-world clips cannot support large-scale training, nor do they allow for parametric diagnostic evaluation to analyze how performance degrades with task complexity.

VideoReasonBench. VideoReasonBench (Liu et al., 2026) also features cups-game tasks, but the swaps are explicitly indicated by arrows overlaid on the frames (Fig. 7). These annotations effectively act as symbolic “swap tokens”, allowing models to deduce state transitions from static in-frame cues. In contrast, VET-Bench presents a more realistic setting where only the moving objects themselves are visible, forcing models to rely on spatiotemporal continuity across frames. Consequently, models like Gemini 2.5 Pro can score 56% on VideoReasonBench but fall to near-chance on VET-Bench.

These limitations of existing benchmarks underscore the need for a controlled diagnostic environment. VET-Bench ensures that once the target is concealed, no single frame reveals either (i) the **target’s identity** or (ii) the **swap operation**. By removing static, frame-level cues, VET-Bench forces VLMs to rely exclusively on fine-grained spatiotemporal perception.

3 Experiments

3.1 Experimental Setup

Models. We evaluate a comprehensive suite of proprietary and open-source VLMs that natively support video input, including Gemini 3 (Pichai et al., 2025), Gemini 2.5 (Comanici et al., 2025), Qwen3.5 (Qwen Team, 2026), Qwen3-VL (Bai et al., 2025a), GLM-4.6V-Flash (GLM-V Team et al., 2025), ERNIE 4.5 (Baidu ERNIE Team, 2025), Doubao-Seed-1.8, Doubao-Seed-2.0 (Bytedance Seed, 2026a,b), Kimi K2.5 (Kimi Team et al., 2026), PerceptionLM (Cho et al., 2025), and Molmo2 (Clark et al., 2026). For each model, we use the maximum supported frame rate, or the default sampling rate where unspecified. Reasoning models are evaluated using their default thinking configurations.

Settings. VLMs adopt different frame sampling strategies. To prevent performance discrepancies arising solely from temporal resolution bottlenecks, we standardize each swap operation to a duration of 2.0 seconds. Under this choice, even a model sampling at 1 FPS observes every swap unambiguously. Consider a 1 FPS sampling of an episode in which each swap takes 2s: the resulting frames alternate between *stable* states, in which all objects occupy their canonical positions, and *transition* states, in which exactly the two exchanged objects are crossing paths. Each transition frame therefore reveals

which pair is being exchanged, and a solver can determine the answer by recording the target’s initial position and, at every transition frame, updating that position only when the target belongs to the exchanged pair. Sampling only the endpoints of each swap would instead yield consecutive stable frames that are indistinguishable for visually identical objects, leaving the answer unrecoverable from the input. In our experiments, open-source VLMs receive denser inputs, typically 2–8 FPS; Gemini 3 samples at 1 FPS by default, and accommodating this sparser rate is precisely why every swap lasts 2 seconds. We evaluate 5 swaps per episode, giving a total duration of 12 seconds (a 2s initial phase followed by a 10s shuffling phase) to balance temporal resolution against context length.

Metrics. All tasks are evaluated in the standard multiple-choice question answering (MCQA) format to facilitate automated grading. Detailed prompts are included in Figs. 10 to 12. Performance is measured via top-1 accuracy. For a task involving N objects, the random baseline is $1/N$.

3.2 Results

The results in Fig. 2 reveal that all evaluated VLMs perform near the random-guessing baseline. This failure is consistent across model sizes and for both reasoning and non-reasoning models. Based on our qualitative analysis of failure modes (Figs. 10 to 12), we categorize the errors into the following three primary patterns:

Direct Answer. Several models, particularly non-reasoning models such as Molmo2, PerceptionLM, and GLM-4.6V-Flash, often output only a final answer without any CoT. Their responses appear to be random guesses.

Coarse Description. Some models successfully identify the initial state but fail to perceive the shuffling phase, instead relying on high-level semantic descriptions (Fig. 12). For example, Qwen3-VL-8B-Instruct generates “*The cups are shuffled in a shell game-like motion, with the ball remaining under one of the cups throughout the video.*” Kimi K2.5 generates “*00:00:02.000 onwards: All three cups begin moving around in a shuffling pattern. 00:00:12.000: The cups return to their original positions.*” Such reasoning collapses fine-grained swap events into a coarse global description, making the final prediction effectively random.

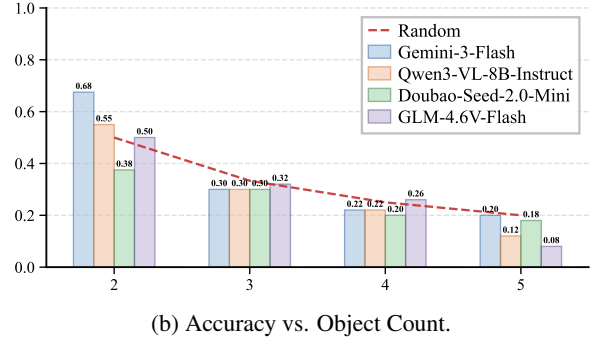
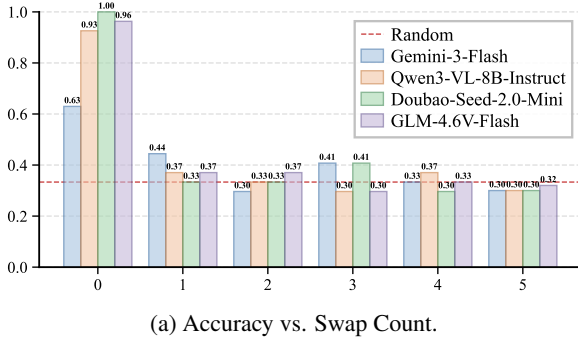


Figure 3: Performance of VLMs under different swap and object counts.

Inaccurate Perception and Hallucination.

Models with stronger reasoning capabilities, such as Gemini 3 Pro, do attempt to generate explicit swap sequences. While the thinking steps are linguistically coherent and logically valid (e.g., “... Move 1 (00:02 - 00:03): The leftmost cup (with the ball) swaps with the middle cup. The ball is now in the middle position. Move 2 ...”), they are often grounded in incorrect visual perceptions: models misidentify which entities are moving or hallucinate swaps that never occur (Figs. 10 and 11). Given that each reasoning step depends on the correctness of the preceding one, any mistake in an intermediate step typically leads to an incorrect final prediction. Symbolic reasoning cannot compensate for failures in pixel-level grounding. Consequently, the final answer remains near the random-guessing baseline.

3.3 Diagnostic Analysis

A key advantage of VET-Bench over real-world video datasets is its programmatic generation, which allows for precise, isolated parameter control. While real-world benchmarks suffer from covarying variables, VET-Bench enables us to systematically scale specific difficulty dimensions to analyze how they affect performance and pinpoint where a model’s visual entity tracking capabilities break down.

Swap Count. We first examine how the number of swap operations per episode affects performance (Fig. 3a). For zero-swap episodes, most models achieve near-perfect accuracy. Models like Doubao-Seed-2.0 Mini can generate the final answer directly without any CoT and achieve perfect scores. This is expected, as the ball’s location is directly observable in the opening frames, mirroring the static frame-level cues. However, performance drops substantially with just one swap, and quickly

converges to the random-guessing baseline thereafter. Interestingly, Gemini 3 Pro and Gemini 3 Flash exhibit a unique failure mode in zero-swap scenarios. Given our standard prompt (“Which cup contains the ball at the end of the video?”), they often correctly identify the initial state but hallucinate a sequence of non-existent swaps, causing an incorrect final prediction (Fig. 13), mirroring the *event hallucination* described by Zhang et al. (2024). When prompted for the ball’s location at the start (“Which cup contains the ball at the start of the video?”), their accuracy returns to near-perfect. This prompt-sensitive hallucination pattern is not observed in other evaluated VLMs.

Object Count. We further analyze performance across varying object counts $N \in \{2, 3, 4, 5\}$ (Fig. 3b). Notably, even at $N = 2$, where the task reduces to a parity problem and the final position depends solely on whether the total number of swaps is even or odd, all evaluated models stay close to the 0.50 random baseline. As N increases, accuracy scales inversely with the number of objects ($1/N$), further indicating that current VLMs do not perform genuine entity tracking but instead resort to random guessing.

Frame Rate. The failures in Section 3.2 could in principle stem from insufficient temporal sampling rather than from a tracking deficit. We therefore evaluate models at various sampling rates. Where the sampling rate is configurable, we set it directly; where it is not, we lengthen each swap proportionally, which scales the frames observed per swap under the model’s default rate. Accuracy remains at the random baseline in all conditions, with no consistent improvement under denser sampling (Table 2). Combined with the observability argument in Section 3.1, this indicates that the models do receive sufficient evidence of each swap and that

the bottleneck lies in perception and state tracking rather than temporal resolution.

4 Theoretical Analysis

The empirical failure of current VLMs on the shell game may stem from practical bottlenecks, such as insufficient training data or lossy visual input encoding. Here, we set aside these factors and study the computational structure of the visual entity tracking task. To examine whether it can be expressed by transformer-based VLMs under the standard language-modeling question-answering paradigm, we formalize it as the following decision problem.

Definition 1 (Visual Entity Tracking, TRACK_k). *TRACK_k is the problem of tracking k visually indistinguishable objects in a video $V = (F_0, \dots, F_T)$ of $T + 1$ frames on an $H \times W$ grid, where k , H , and W are constants. The input is assumed to satisfy localization and continuity conditions given in Appendix B. Let π be the global permutation that maps the k objects from their initial lexicographic ordering of positions in frame 0 to their final ordering in frame T . The problem asks whether π is the identity permutation.*

Drawing inspiration from state-tracking problems (Merrill et al., 2024), we characterize the complexity of TRACK_k via the word problem for the symmetric group S_5 , a canonical NC^1 -complete problem (Barrington, 1989).

Definition 2 (Word Problem for S_5 , WORD_{S_5}). *Let the generators be the adjacent transpositions $\tau_j = (j, j + 1)$ for $j \in \{1, 2, 3, 4\}$. Given a sequence $W = \langle \sigma_1, \dots, \sigma_N \rangle$ where each $\sigma_i \in \{\tau_1, \tau_2, \tau_3, \tau_4\}$, let $\Pi = \sigma_N \circ \dots \circ \sigma_1$. The output is true iff Π is the identity.*

Theorem 1. *For any fixed $k \geq 5$, TRACK_k is NC^1 -complete.*

Proof Sketch. Membership in NC^1 follows by computing, for each adjacent pair of frames, the unique permutation $\pi_t \in S_k$ induced by the localization and continuity conditions, and then composing $\pi = \pi_{T-1} \circ \dots \circ \pi_0$ in NC^1 (Lemma 1). Hardness is established via reduction from WORD_{S_5} by constructing a video that physically realizes adjacent transposition generators (Lemma 2). Full proofs are provided in Appendix B. $\square$

Expressivity and the Role of CoT. Prior work shows that constant-depth transformers can be simulated within the circuit class TC^0 (Merrill and

Sabharwal, 2023, 2024). Assuming the widely held conjecture that $\text{TC}^0 \subsetneq \text{NC}^1$, fixed-depth transformers are theoretically incapable of solving NC^1 -complete problems like TRACK_k for $k \geq 5$ on arbitrary-length sequences. The NC^1 -completeness result therefore provides a theoretical foundation for the necessity of CoT in visual entity tracking (Feng et al., 2023; Li et al., 2024d; Huang et al., 2025).

If objects possess unique visual identifiers, the task collapses from a sequential state-tracking problem into a parallelizable visual search problem (AC^0). This explains why tracking objects with distinct appearances is often easier: models can rely on appearance-based rematching rather than entity tracking. This is analogous to the difference in hardness between source-target notation (UCI) and standard algebraic notation (SAN) (Toshniwal et al., 2022) for chess state tracking, as discussed by Merrill et al. (2024).

While humans often perceive visual entity tracking as an intuitive perceptual ability, our analysis suggests it belongs to the same complexity class as other state-tracking tasks, such as tracking chess moves or narrative entities (Merrill et al., 2024), typically considered difficult due to their latent reasoning requirements. We posit that humans manage this complexity through foveal eye movements, which can be viewed as a form of physical CoT by explicitly updating the target’s state over time.

Learnability. Barrington showed that the word problem for every fixed non-solvable finite group is NC^1 -complete under AC^0 reductions (Barrington, 1989). Since S_k is non-solvable for $k \geq 5$, this explains the $k \geq 5$ threshold in our hardness result. For smaller groups such as S_2 , the task reduces to parity and lies in TC^0 , which can potentially be expressed by fixed-depth transformers. However, expressivity does not imply learnability. Prior work on parity and coin-flip tasks shows that CoT is often crucial for effective training and length generalization (Wei et al., 2022; Anil et al., 2022; Wies et al., 2023; Kim and Suzuki, 2025). Thus, smaller cases still motivate CoT from a learnability perspective.

To examine whether a VLM can learn visual entity tracking from direct-answer supervision without CoT, we train Qwen2.5-VL-3B-Instruct (Bai et al., 2025b) on 500 synthesized cups-game videos (varying only in shuffle permutations) with an 8 FPS sampling rate. As shown in Fig. 9, the loss re-

mains stagnant at the level of random chance even after 60 epochs. This phenomenon mirrors the difficulty of learning the parity task (Hahn and Rofin, 2024; Kim and Suzuki, 2025). Although parity is in TC^0 and therefore expressible by transformers, the target label lacks low-order statistical correlations with input features, making it difficult to train end-to-end using gradient-based algorithms. In our case, the long sequence of video tokens makes it difficult for the model to map the input directly to the final label. Consequently, the model fails to capture the underlying dynamics and instead resorts to predicting based on label frequency within the training distribution.

5 Spatiotemporal Grounded CoT

The preceding analysis motivates additional intermediate computation for general visual entity tracking. While models like Gemini 3 Pro produce logically valid CoT, they still perform near-random on VET-Bench due to inaccurate perceptual grounding (Figs. 10 and 11). To address this, we propose **Spatiotemporal Grounded Chain-of-Thought (SG-CoT)**: generating object trajectories as CoT. We leverage Molmo2 (Clark et al., 2026) for this task, as it is extensively trained on video object tracking and exhibits strong grounding capabilities that can be repurposed for visual entity tracking QA.

5.1 Limitations of the Base Model

Molmo2 exhibits different failure modes under direct and CoT prompting.

Direct Prompting. Given a standard query such as “Where is the Queen of Hearts at the end of the video?”, Molmo2 typically outputs only a final answer (e.g., “Left”) without explicitly tracking the target over time, resulting in near-random performance (Fig. 14).

CoT Prompting. When explicitly instructed to track the target before answering (e.g., “Track the object and then answer...”), Molmo2 instead produces the structured tracking format learned during pretraining:

```
<tracks coords="timestamp object_idx x
y;...">Object</tracks>
```

Here, timestamps are spaced at 0.5-second intervals, and (x, y) denotes coordinates normalized to the range $[0, 1000]$. Although the generated trajectory provides the intermediate information needed

to solve the task, the base model stops after producing the trajectory and does not effectively use it as an intermediate step to derive the final answer.

These observations suggest that Molmo2 inherently possesses a strong object-tracking capability, but this capability is not reliably elicited by prompting and is not naturally integrated into the reasoning process.

5.2 Training

We propose a simple and efficient approach that trains Molmo2-SGCoT to solve VET-Bench tasks end-to-end using a small number of text-only samples without video input. We adopt CoT prompting (e.g., by prefixing the query with “Track [object]”) to elicit tracking behavior (Fig. 15). A training example for Molmo2-SGCoT, including the prompt and response, is shown below:

Track the Queen of Hearts and answer where is the Queen of Hearts at the end of the video?

```
<tracks coords="0.0 1 745 512;0.5 1
745 512;1.0 1 745 500;1.5 1 742
425;2.0 1 681 296; ... 12.0 1 247
500">the Queen of Hearts</tracks>
Answer: left.
```

In VET-Bench, the final answer (“left”, “middle”, or “right”) depends only on the terminal location of the tracked object, which is explicitly specified by the final coordinates (shown in red). During training, we mask the loss on all synthesized trajectory tokens within `<tracks></tracks>` and supervise only the final answer (underlined), encouraging the model to preserve its grounding capability while learning to derive the final answer from the SGCoT. Notably, our method requires no explicit training on VET-Bench videos: the trajectories can be synthesized programmatically with a simple Python script. While randomly generated trajectories that respect the expected format and answer mapping are sufficient for alignment, we find that using in-distribution trajectories minimizes degradation of the model’s tracking ability. Our SGCoT alignment is highly efficient; one epoch over 300 samples takes 3 minutes on a single A100 GPU, improving Molmo2’s accuracy from near-random to 91.4% on VET-Bench. Training details are provided in Table 4.

5.3 Discussion

Error Analysis. Despite its high accuracy, Molmo2-SGCoT still fails on a small fraction of episodes. As shown in Fig. 16, errors typically arise in the SGCoT perception stage, when the model confuses two visually identical objects mid-shuffle. These failures appear as abrupt “jumps” in the emitted trajectory, either spatial (the track teleports to a different object) or temporal (timestamps are skipped), and a corrupted trajectory yields an incorrect terminal coordinate and therefore an incorrect answer. We did not observe failures in which a correct trajectory was paired with a wrong final answer, indicating that the trajectory-to-answer mapping learned during alignment is reliable.

Object Tracking as a Prerequisite. SGCoT is not a model-agnostic prompting strategy. It relies on the underlying VLM being able to generate accurate object trajectories. Applying SGCoT-style prompting to models without reliable video object-tracking capability yields little improvement (Table 3). If the generated trajectory is incorrect, subsequent reasoning over that trajectory cannot recover the correct answer. Among the VLMs evaluated in our experiments, Molmo2 is the only model that reliably produces precise spatiotemporal trajectories at designated timestamps, consistent with its explicit training for video object tracking. Prompting cannot substitute for a capability the base model does not have; SGCoT converts an existing perceptual capability into a reasoning step instead of creating one.

Why Does SGCoT Help? We attribute the effectiveness of SGCoT to its fine-grained representation of intermediate states. In particular, it explicitly aligns (i) when an event occurs (using discrete timestamps) and (ii) where each entity is located (using exact spatial coordinates). This combination yields an unambiguous state representation.

In contrast, many generic descriptive CoTs produced by other VLMs, as shown in Figs. 10 to 12, are loosely grounded in time and space. CoTs that ground events with coarse time ranges (e.g., “00:01–00:02”) can be brittle; motion events often have non-integer start and end times. These coarse intervals lead to temporal misalignment (Du et al., 2025), causing errors to accumulate. Similarly, CoTs that rely on coarse referring expressions (e.g., “left”, “right”) underspecify the states. SGCoT mitigates these ambiguities by discretizing time at con-

sistent intervals and ensuring that a precise spatial state corresponds to each timestamp.

6 Related Work

Video Benchmarks for VLMs. Existing video QA benchmarks (Zhou et al., 2025; Song et al., 2025; Zhao et al., 2025; Fang et al., 2024; Li et al., 2024a,b) mainly focus on general video understanding. However, many fail to decouple temporal dynamics from static cues, enabling VLMs to achieve high performance via shortcuts (Lei et al., 2023). Recent benchmarks—such as TempCompass (Liu et al., 2024b), TemporalBench (Cai et al., 2024), E.T. Bench (Liu et al., 2024a), TVBench (Cores et al., 2025), VideoVista (Li et al., 2024c), TOMATO (Shangguan et al., 2025), MotionBench (Hong et al., 2025), TUNA (Kong et al., 2025), Know-Show (Sugandhika et al., 2025), EgoTempo (Plizzari et al., 2025), VCR-Bench (Qi et al., 2025), MVP (Krojer et al., 2025), AoTBench (Xue et al., 2025), SpookyBench (Upadhyay et al., 2026), Spatial4D-Bench (Wang et al., 2026b), TimeBlind (Li et al., 2026), NarrativeTrack (Ha et al., 2026)—expose the limitation that VLMs often rely on single-frame biases without genuinely modeling temporal dynamics. Following this line of work, our VET-Bench targets fine-grained spatiotemporal perception and reasoning.

Reasoning with Grounding. Recent VLMs (Clark et al., 2026; Deitke et al., 2025) have demonstrated strong grounding abilities through targeted training (Lin et al., 2014; Yu et al., 2016). This facilitates reasoning with grounding to improve robustness and reliability (Jiang et al., 2026; Li et al., 2025; Chen et al., 2026; Wang et al., 2026a; Wu et al., 2026b; Gu et al., 2025; Sarch et al., 2025). Grounded CoT (Wu et al., 2026a; Xia et al., 2025) explicitly injects bounding boxes into CoT to enhance faithfulness and reduce hallucination. We extend this to the temporal domain and show that SGCoT can reliably solve visual entity tracking.

Entity Tracking. Entity tracking has been studied in textual domains (Toshniwal et al., 2022; Kim and Schuster, 2023), with a focus on evaluating LLMs’ abilities to maintain representations of entity states. Merrill et al. (2024) prove that chess state tracking in UCI notation is NC^1 -complete. Building on their constructive proof, we show that visual entity tracking also falls into this class and necessitates CoT for generalization. MET-Bench

(Cohen and Mooney, 2026) studies entity tracking in the shell game and chess using image and text inputs. In contrast, our work addresses the visual entity tracking problem in videos.

7 Conclusion

In this work, we identified visual entity tracking as a fundamental bottleneck in current VLMs. We introduced VET-Bench and demonstrated that state-of-the-art models perform near chance when appearance cues are removed. Our theoretical analysis established that visual entity tracking is NC¹-complete, suggesting that fixed-depth VLMs are inherently limited in expressivity when solving such tasks without CoT. We empirically verified that VLMs are unable to solve the shell game in direct-answer mode. Finally, by aligning Molmo2 to generate SGCoT before the final answer, we achieved state-of-the-art accuracy exceeding 90% on VET-Bench. Ultimately, we hope this work paves the way for future VLMs to leverage SGCoT in broader video perception and reasoning.

Limitations

We study a simplified setting in VET-Bench where the final answer is fully determined by the location information encoded in the generated SGCoT. In real-world scenarios, however, answering the question may also require additional visual perception within the final frames, for example, to resolve general referring expressions (“Which cup contains the ball at the end of the video, from the person’s point of view?”). Future work could explore integrating SGCoT with complex referring expressions across arbitrary frames. Moreover, our analysis assumes localization and continuity conditions to ensure that swap movements remain unambiguous. In practice, tracking is often more challenging: objects move in close proximity, overlap, exhibit motion blur, or undergo occlusion (Qi et al., 2022; Pothiraj et al., 2025). Handling these complex scenarios may require incorporating models with stronger physical priors or world models to enable more robust tracking.

References

Cem Anil, Yuhuai Wu, Anders Andreassen, Aitor Lewkowycz, Vedant Misra, Vinay Ramasesh, Ambrose Slone, Guy Gur-Ari, Ethan Dyer, and Behnam Neyshabur. 2022. Exploring length generalization in

large language models. *Advances in Neural Information Processing Systems*, 35:38546–38556.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhi-fang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, and 45 others. 2025a. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025b. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*.

Baidu ERNIE Team. 2025. [ERNIE 4.5 technical report](#).

David A Barrington. 1989. Bounded-width polynomial-size branching programs recognize exactly those languages in NC1. *Journal of Computer and System Sciences*, 38(1):150–164.

Bytedance Seed. 2026a. [Seed1.8 model card: Towards generalized real-world agency](#). *Preprint*, arXiv:2603.20633.

Bytedance Seed. 2026b. [Seed2.0 model card: Towards intelligence frontier for real-world complexity](#). *Preprint*, arXiv:2607.00248.

Mu Cai, Reuben Tan, Jianrui Zhang, Bocheng Zou, Kai Zhang, Yao Feng, Fangrui Zhu, Jing Gu, Yiwu Zhong, Yuzhang Shang, Yao Dou, Jaden Park, Jianfeng Gao, Yong Jae Lee, and Jianwei Yang. 2024. [TemporalBench: Benchmarking fine-grained temporal understanding for multimodal video models](#). In *Workshop on Video-Language Models @ NeurIPS 2024*.

Yiming Chen, Zekun Qi, Wenyao Zhang, Xin Jin, Li Zhang, and Peidong Liu. 2026. [Reasoning in space via grounding in the world](#). In *The Fourteenth International Conference on Learning Representations*.

Jang Hyun Cho, Andrea Madotto, Effrosyni Mavroudi, Triantafyllos Afouras, Tushar Nagarajan, Muhammad Maaz, Yale Song, Tengyu Ma, Shuming Hu, Suyog Jain, Miguel Martin, Huiyu Wang, Hanoona Abdul Rasheed, Peize Sun, Po-Yao Huang, Daniel Bolya, Nikhila Ravi, Shashank Jain, Tammy Stark, and 10 others. 2025. [PerceptionLM: Open-access data and models for detailed visual understanding](#). In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Christopher Clark, Jieyu Zhang, Zixian Ma, Jae Sung Park, Rohun Tripathi, Sangho Lee, Mohammadreza Salehi, Jason Ren, Chris Dongjoo Kim, Yinyu Yang, Vincent Shao, Yue Yang, Weikai Huang, Ziqi Gao, Taira Anderson, Jianrui Zhang, Jitesh Jain, George Stoica, Ali Farhadi, and Ranjay Krishna. 2026.

- Molmo2: Open weights and data for vision-language models with video understanding and grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 28652–28668.
- Vanya Cohen and Ray Mooney. 2026. [MET-Bench: Multimodal entity tracking for evaluating the limitations of vision-language and reasoning models](#). In *Forty-third International Conference on Machine Learning*.
- Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, Luke Marris, Sam Petulla, Colin Gaffney, Asaf Aharoni, Nathan Lintz, Tiago Cardal Pais, Henrik Jacobsson, Idan Szpektor, Nan-Jiang Jiang, and 3416 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.
- Daniel Cores, Michael Dorkenwald, Manuel Mucientes, Cees G. M. Snoek, and Yuki M. Asano. 2025. Lost in time: A new temporal benchmark for VideoLLMs. In *British Machine Vision Conference (BMVC)*.
- Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, Jiasen Lu, Taira Anderson, Erin Bransom, Kiana Ehsani, Huang Ngo, YenSung Chen, Ajay Patel, Mark Yatskar, Chris Callison-Burch, and 31 others. 2025. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 91–104.
- Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient finetuning of quantized LLMs. *Advances in neural information processing systems*, 36:10088–10115.
- Hao Du, Bo Wu, Yan Lu, and Zhendong Mao. 2025. SVLTA: Benchmarking vision-language temporal alignment via synthetic video situation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13798–13809.
- Xinyu Fang, Kangrui Mao, Haodong Duan, Xiangyu Zhao, Yining Li, Dahua Lin, and Kai Chen. 2024. MMBench-Video: A long-form multi-shot benchmark for holistic video understanding. *Advances in Neural Information Processing Systems*, 37:89098–89124.
- Guhaog Feng, Bohang Zhang, Yuntian Gu, Haotian Ye, Di He, and Liwei Wang. 2023. Towards revealing the mystery behind chain of thought: a theoretical perspective. *Advances in Neural Information Processing Systems*, 36:70757–70798.
- Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiwu Zheng, Enhong Chen, Caifeng Shan, and 2 others. 2025. Video-MME: The first-ever comprehensive evaluation benchmark of multi-modal LLMs in video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24108–24118.
- Pascale Fung, Yoram Bachrach, Asli Celikyilmaz, Kamalika Chaudhuri, Delong Chen, Willy Chung, Emmanuel Dupoux, Hongyu Gong, Hervé Jégou, Alessandro Lazaric, Arjun Majumdar, Andrea Madotto, Franziska Meier, Florian Metze, Louis-Philippe Morency, Théo Moutakanni, Juan Pino, Basile Terver, Joseph Tighe, and 2 others. 2025. Embodied AI agents: Modeling the world. *arXiv preprint arXiv:2506.22355*.
- Rohit Girdhar and Deva Ramanan. 2020. [CATER: A diagnostic dataset for compositional actions & temporal reasoning](#). In *International Conference on Learning Representations*.
- GLM-V Team, Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, Shuaiqi Duan, Weihang Wang, Yan Wang, Yean Cheng, Zehai He, Zhe Su, Zhen Yang, Ziyang Pan, and 74 others. 2025. GLM-4.5V and GLM-4.1V-Thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. *arXiv preprint arXiv:2507.01006*.
- Xin Gu, Haoji Zhang, Qihang Fan, Jingxuan Niu, Zhipeng Zhang, Libo Zhang, Guang Chen, Fan Chen, Longyin Wen, and Sijie Zhu. 2025. Thinking with bounding boxes: Enhancing spatio-temporal video grounding via reinforcement fine-tuning. *arXiv preprint arXiv:2511.21375*.
- Hyeonjeong Ha, Jinjin Ge, Bo Feng, Kaixin Ma, and Gargi Chakraborty. 2026. NarrativeTrack: Evaluating entity-centric reasoning for narrative understanding. *arXiv preprint arXiv:2601.01095*.
- Michael Hahn and Mark Rojin. 2024. Why are sensitive functions hard for transformers? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14973–15008.
- Wenyi Hong, Yean Cheng, Zhuoyi Yang, Weihang Wang, Lefan Wang, Xiaotao Gu, Shiyu Huang, Yuxiao Dong, and Jie Tang. 2025. MotionBench: Benchmarking and improving fine-grained video motion understanding for vision language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 8450–8460.
- Kairui Hu, Penghao Wu, Fanyi Pu, Wang Xiao, Xiang Yue, Bo Li, Yuanhan Zhang, and Ziwei Liu. 2026. [Video-MMMU: Evaluating knowledge acquisition from multidisciplinary professional videos](#). In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long*

- Papers), pages 27798–27828, San Diego, California, United States. Association for Computational Linguistics.
- Yu Huang, Zixin Wen, Aarti Singh, Yuejie Chi, and Yuxin Chen. 2025. [Transformers provably learn chain-of-thought reasoning with length generalization](#). In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.
- Kelly Jaakkola. 2014. [Do animals understand invisible displacement? A critical review](#). *Journal of Comparative Psychology*, 128(3):225–239.
- Qing Jiang, Xingyu Chen, Zhaoyang Zeng, Junzhi Yu, and Lei Zhang. 2026. [Rex-Thinker: Grounded object referring via chain-of-thought reasoning](#). In *The Fourteenth International Conference on Learning Representations*.
- Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. 2017. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2901–2910.
- Juno Kim and Taiji Suzuki. 2025. [Transformers provably solve parity efficiently with chain of thought](#). In *The Thirteenth International Conference on Learning Representations*.
- Najoung Kim and Sebastian Schuster. 2023. Entity tracking in language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3835–3855.
- Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, S. H. Cai, Yuan Cao, Ziwei Chai, Y. Charles, H. S. Che, Cheng Chen, Guanduo Chen, Huarong Chen, Jia Chen, Jianlong Chen, Jun Chen, Kefan Chen, Liang Chen, Ruijue Chen, Xinhao Chen, and 318 others. 2026. Kimi K2.5: Visual agentic intelligence. *arXiv preprint arXiv:2602.02276*.
- Fanheng Kong, Jingyuan Zhang, Hongzhi Zhang, Shi Feng, Daling Wang, Linhao Yu, Xingguang Ji, Yu Tian, Victoria W., and Fuzheng Zhang. 2025. [TUNA: Comprehensive fine-grained temporal understanding evaluation on dense dynamic videos](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1810–1839, Vienna, Austria. Association for Computational Linguistics.
- Benno Kroyer, Mojtaba Komeili, Candace Ross, Quentin Garrido, Koustuv Sinha, Nicolas Ballas, and Mido Assran. 2025. [A shortcut-aware video-QA benchmark for physical understanding via minimal video pairs](#). *Transactions on Machine Learning Research*.
- Jie Lei, Tamara Berg, and Mohit Bansal. 2023. Revealing single frame bias for video-and-language learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 487–507.
- Baiqi Li, Kangyi Zhao, Ce Zhang, Chancharik Mitra, Jean de Dieu Nyandwi, and Gedas Bertasius. 2026. TimeBlind: A spatio-temporal compositionality benchmark for video LLMs. *arXiv preprint arXiv:2602.00288*.
- Bohao Li, Yuying Ge, Yixiao Ge, Guangzhi Wang, Rui Wang, Ruimao Zhang, and Ying Shan. 2024a. SEED-Bench: Benchmarking multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13299–13308.
- 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.
- Yunxin Li, Xinyu Chen, Baotian Hu, Longyue Wang, Haoyuan Shi, and Min Zhang. 2024c. VideoVista: A versatile benchmark for video understanding and reasoning. *arXiv preprint arXiv:2406.11303*.
- Zejun Li, Ruipu Luo, Jiwen Zhang, Minghui Qiu, Xuan-Jing Huang, and Zhongyu Wei. 2025. VoCoT: Unleashing visually grounded multi-step reasoning in large multi-modal models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3769–3798.
- Zhiyuan Li, Hong Liu, Denny Zhou, and Tengyu Ma. 2024d. Chain of thought empowers transformers to solve inherently serial problems. In *The Twelfth International Conference on Learning Representations*.
- Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft COCO: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.
- Ye Liu, Zongyang Ma, Zhongang Qi, Yang Wu, Ying Shan, and Chang W Chen. 2024a. E.T. Bench: Towards open-ended event-level video-language understanding. *Advances in Neural Information Processing Systems*, 37:32076–32110.
- 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.
- Yuanxin Liu, Kun Ouyang, Haoning Wu, Yi Liu, Lin Sui, Xinhao Li, Yan Zhong, Y. Charles, Xinyu Zhou, and Xu Sun. 2026. [VideoReasonBench: Can](#)

- MLLMs perform vision-centric complex video reasoning? In *The Fourteenth International Conference on Learning Representations*.
- Loïc Magne, Anas Awadalla, Guanzhi Wang, Yinzhen Xu, Joshua Belofsky, Fengyuan Hu, Joohwan Kim, Ludwig Schmidt, Georgia Gkioxari, Jan Kautz, Yisong Yue, Yejin Choi, Yuke Zhu, and Linxi Fan. 2026. NitroGen: An open foundation model for generalist gaming agents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21511–21521.
- William Merrill, Jackson Petty, and Ashish Sabharwal. 2024. The illusion of state in state-space models. In *International Conference on Machine Learning*, pages 35492–35506. PMLR.
- William Merrill and Ashish Sabharwal. 2023. The parallelism tradeoff: Limitations of log-precision transformers. *Transactions of the Association for Computational Linguistics*, 11:531–545.
- William Merrill and Ashish Sabharwal. 2024. [The expressive power of transformers with chain of thought](#). In *The Twelfth International Conference on Learning Representations*.
- Viorica Patraucean, Lucas Smaira, Ankush Gupta, Adria Recasens Contente, Larisa Markeeva, Dylan Sunil Banarse, Skanda Koppula, Joseph Heyward, Mateusz Malinowski, Yi Yang, Carl Doersch, Tatiana Matejovicova, Yury Sulsky, Antoine Miech, Alexandre Fréchet, Hanna Klimczak, Raphael Koster, Junlin Zhang, Stephanie Winkler, and 5 others. 2023. [Perception Test: A diagnostic benchmark for multi-modal video models](#). In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
- Sundar Pichai, Demis Hassabis, and Koray Kavukcuoglu. 2025. [A new era of intelligence with Gemini 3](#). Google Blog. Accessed 2025-12-12.
- Chiara Plizzari, Alessio Tonioni, Yongqin Xian, Achin Kulshrestha, and Federico Tombari. 2025. Omnia de EgoTempo: Benchmarking temporal understanding of multi-modal LLMs in egocentric videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24129–24138.
- Atin Pothiraj, Elias Stengel-Eskin, Jaemin Cho, and Mohit Bansal. 2025. CAPTURE: Evaluating spatial reasoning in vision language models via occluded object counting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8001–8010.
- Jiyang Qi, Yan Gao, Yao Hu, Xinggang Wang, Xiaoyu Liu, Xiang Bai, Serge Belongie, Alan Yuille, Philip HS Torr, and Song Bai. 2022. Occluded video instance segmentation: A benchmark. *International Journal of Computer Vision*, 130(8):2022–2039.
- Yukun Qi, Yiming Zhao, Yu Zeng, Xikun Bao, Wenxuan Huang, Lin Chen, Zehui Chen, Jie Zhao, Zhongang Qi, and Feng Zhao. 2025. VCR-Bench: A comprehensive evaluation framework for video chain-of-thought reasoning. *arXiv preprint arXiv:2504.07956*.
- Qwen Team. 2026. [Qwen3.5: Towards native multi-modal agents](#).
- Gabriel Herbert Sarch, Snigdha Saha, Naitik Khandelwal, Ayush Jain, Michael J. Tarr, Aviral Kumar, and Katerina Fragkiadaki. 2025. [Grounded reinforcement learning for visual reasoning](#). In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.
- Ziyao Shangguan, Chuhan Li, Yuxuan Ding, Yanan Zheng, Yilun Zhao, Tesca Fitzgerald, and Arman Cohan. 2025. [TOMATO: Assessing visual temporal reasoning capabilities in multimodal foundation models](#). In *The Thirteenth International Conference on Learning Representations*.
- SIMA Team, Adrian Bolton, Alexander Lerchner, Alexandra Cordell, Alexandre Moufaret, Andrew Bolt, Andrew Lampinen, Anna Mitenkova, Arne Olav Hallingstad, Bojan Vujatovic, Bonnie Li, Cong Lu, Daan Wierstra, Daniel P. Sawyer, Daniel Slater, David Reichert, Davide Vercelli, Demis Hassabis, Drew A. Hudson, and 47 others. 2025. SIMA 2: A generalist embodied agent for virtual worlds. *arXiv preprint arXiv:2512.04797*.
- Enxin Song, Wenhao Chai, Weili Xu, Jianwen Xie, Yuxuan Liu, and Gaoang Wang. 2025. Video-MMLU: A massive multi-discipline lecture understanding benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 6158–6172.
- Chinthani Sugandhika, Chen Li, Deepu Rajan, and Basura Fernando. 2025. Know-Show: Benchmarking video-language models on spatio-temporal grounded reasoning. *arXiv preprint arXiv:2512.05513*.
- Jacob Thompson, Emiliano Garcia-Lopez, and Yonatan Bisk. 2025. [REM: Evaluating LLM embodied spatial reasoning through multi-frame trajectories](#). In *Second Conference on Language Modeling*.
- Shubham Toshniwal, Sam Wiseman, Karen Livescu, and Kevin Gimpel. 2022. Chess as a testbed for language model state tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11385–11393.
- Ujjwal Upadhyay, Mukul Ranjan, Zhiqiang Shen, and Mohamed Elhoseiny. 2026. Time blindness: Why video-language models can’t see what humans can? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 30906–30918.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

- Haochen Wang, Xiangtai Li, Zilong Huang, Anran Wang, Jiacong Wang, Tao Zhang, Jiani Zheng, Sule Bai, Zijian Kang, Jiashi Feng, Zhuochen Wang, and Zhaoxiang Zhang. 2026a. [Traceable evidence enhanced visual grounded reasoning: Evaluation and method](#). In *The Fourteenth International Conference on Learning Representations*.
- Pan Wang, Yang Liu, Guile Wu, Eduardo R. Corral-Soto, Chengjie Huang, Binbin Xu, Dongfeng Bai, Xu Yan, Yuan Ren, Xingxin Chen, Yizhe Wu, Tao Huang, Wenjun Wan, Xin Wu, Pei Zhou, Xuyang Dai, Kangbo Lv, Hongbo Zhang, Yosef Fried, and 7 others. 2026b. Spatial4D-Bench: A versatile 4D spatial intelligence benchmark. *arXiv preprint arXiv:2601.00092*.
- Weihan Wang, Zehai He, Wenyi Hong, Yean Cheng, Xiaohan Zhang, Ji Qi, Ming Ding, Xiaotao Gu, Shiyu Huang, Bin Xu, Yuxiao Dong, and Jie Tang. 2025. LVBench: An extreme long video understanding benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22958–22967.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.
- Noam Wies, Yoav Levine, and Amnon Shashua. 2023. [Sub-task decomposition enables learning in sequence to sequence tasks](#). In *The Eleventh International Conference on Learning Representations*.
- Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. 2024. LongVideoBench: A benchmark for long-context interleaved video-language understanding. *Advances in Neural Information Processing Systems*, 37:28828–28857.
- Qiong Wu, Xiangcong Yang, Yiyi Zhou, Chenxin Fang, Baiyang Song, Xiaoshuai Sun, and Rongrong Ji. 2026a. Grounded chain-of-thought for multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 33577–33587.
- Xiyang Wu, Zongxia Li, Jihui Jin, Gouthaman KV, Vishnu Raj, Nilotpal Sinha, Jingxi Chen, Fan Du, and Dinesh Manocha. 2026b. MASS: Motion-aware spatial-temporal grounding for physics reasoning and comprehension in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pages 9433–9443.
- Jiaer Xia, Bingkui Tong, Yuhang Zang, Rui Shao, and Kaiyang Zhou. 2025. Bootstrapping grounded chain-of-thought in multimodal LLMs for data-efficient model adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 208–217.
- Zihui Xue, Mi Luo, and Kristen Grauman. 2025. [Seeing the arrow of time in large multimodal models](#). In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.
- Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B. Tenenbaum. 2020. [CLEVRER: Collision events for video representation and reasoning](#). In *International Conference on Learning Representations*.
- Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. 2016. Modeling context in referring expressions. In *European conference on computer vision*, pages 69–85. Springer.
- Jiacheng Zhang, Yang Jiao, Shaoxiang Chen, Na Zhao, Zhiyu Tan, Hao Li, Xingjun Ma, and Jingjing Chen. 2024. EventHallusion: Diagnosing event hallucinations in video LLMs. *arXiv preprint arXiv:2409.16597*.
- Yilun Zhao, Haowei Zhang, Lujing Xie, Tongyan Hu, Guo Gan, Yitao Long, Zhiyuan Hu, Weiyuan Chen, Chuhan Li, Zhijian Xu, Chengye Wang, Ziyao Shang-guan, Zhenwen Liang, Yixin Liu, Chen Zhao, and Arman Cohan. 2025. Mmvu: Measuring expert-level multi-discipline video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8475–8489.
- Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Zhengyang Liang, Shitao Xiao, Minghao Qin, Xi Yang, Yongping Xiong, Bo Zhang, Tiejun Huang, and Zheng Liu. 2025. MLVU: Benchmarking multi-task long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13691–13701.

A Detailed Audit of the Perception Test



Figure 4: Example frames from videos involving distinct cups in the Perception Test.



Figure 5: Example frames from videos involving transparent cups in the Perception Test.



Figure 6: Visual shortcuts in videos #7632 and #9896. These videos contain unedited cut frames at the end that reveal the answer by showing the cups being lifted. This unintended disclosure provides a shortcut that allows the model to simply look at the final state to identify the object's location, making the tracking of the shuffle irrelevant.

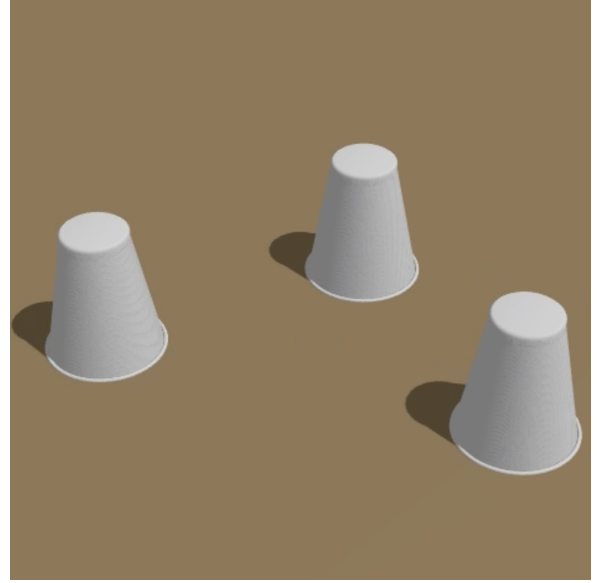
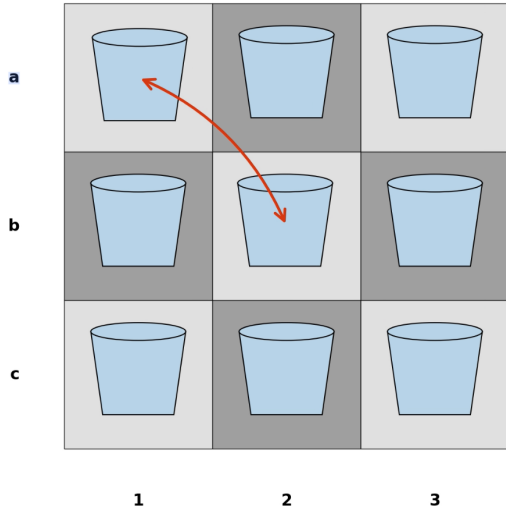


Figure 7: Comparison of the cups-game tasks in VideoReasonBench (left) and VET-Bench (right). VideoReasonBench provides explicit frame-level cues highlighting swap operations (indicated by red arrows), which are absent in VET-Bench.

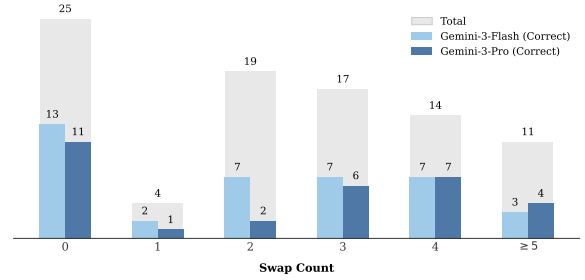
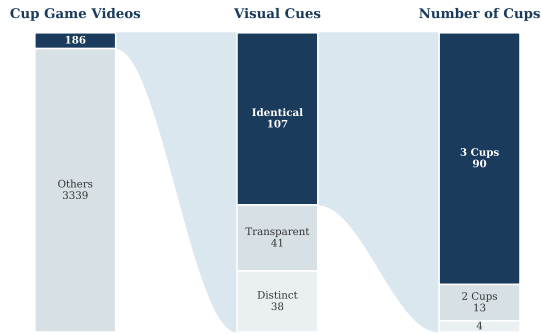


Figure 8: Audit of the Perception Test. Left: Filtering process for the identical 3-cup setting. Right: Distribution of swap counts in the filtered subset, with the number of correct responses from Gemini 3 Pro and Gemini 3 Flash.

Model	Filtered	VET-Bench
<i>Random</i>	0.33	0.33
Gemini 3 Pro	0.31	0.36
Gemini 3 Flash	0.40	0.34
Qwen3-VL-8B-Instruct	0.34	0.34
GLM-4.6V-Flash	0.42	0.29
Doubao-Seed-2.0 Mini	0.40	0.31

Table 1: Performance comparison on the filtered subset and VET-Bench.

B Proof of Theorem 1

Recall that the decision problem TRACK_k takes as input a video $V = (F_0, \dots, F_T)$ of $T+1$ frames on an $H \times W$ pixel grid, where k , H , and W are constants and the input length is $n = \Theta(T)$, and asks whether the permutation π from frame 0 to frame T is the identity.

The input is assumed to satisfy the following conditions:

1. **Localization.** Each frame F_t contains exactly k object centers (grid coordinates), denoted as $C_t = \{c_{t,1}, \dots, c_{t,k}\}$ for $t \in \{0, \dots, T\}$, where $c_{t,i} \in [H] \times [W]$, and the k object centers can be recovered from each frame in NC^1 .
2. **Continuity.** Each object center moves by at most a fixed distance d between consecutive frames, and any two centers within any single frame are at distance $> 2d$, where, for convenience, we take $\text{dist}(\cdot, \cdot)$ to be the L_1 (Manhattan) distance.

The conditions above are standard prerequisites to ensure that the problem is well-posed in controlled environments and are easily satisfied in practice. The localization condition ensures that object centers are efficiently recoverable; for example, using constant-size visual markers allows template matching in NC^1 . The continuity condition formalizes the constraint discussed earlier (Sections 2.1 and 3.1) to prevent ambiguity by guaranteeing a unique correspondence between objects in consecutive frames. Practically, this can be achieved by using a sufficiently high frame rate relative to object speeds, so that objects move only a small distance between adjacent frames.

Lemma 1. $\text{TRACK}_k \in \text{NC}^1$.

Proof. We show that the permutation π can be computed in NC^1 .

For each frame t , let the localization procedure output the set of k distinct centers as grid coordinates. Since H and W are constants, each coordinate has constant bit-length. Hence, per-frame computations are constant-size.

Let $\hat{c}_{t,1}, \dots, \hat{c}_{t,k}$ be the centers sorted lexicographically by (x, y) (i.e., by x coordinate, breaking ties by y). Since both k and the coordinate length are constant, this sorting is computable by a constant-size circuit for each frame.

By the continuity condition, each $\hat{c}_{t,i}$ has at least one successor $\hat{c}_{t+1,j}$ within distance d . Such a j is unique. Suppose there exist distinct indices $j \neq j'$ such that

$$\text{dist}(\hat{c}_{t,i}, \hat{c}_{t+1,j}) \leq d$$

and

$$\text{dist}(\hat{c}_{t,i}, \hat{c}_{t+1,j'}) \leq d,$$

then

$$\text{dist}(\hat{c}_{t+1,j}, \hat{c}_{t+1,j'}) \leq 2d,$$

contradicting the continuity condition in frame $t+1$. Define $\pi_t : [k] \rightarrow [k]$ by $\pi_t(i) = j$.

π_t is injective: Suppose $\pi_t(i) = \pi_t(i') = j$ for some $i \neq i'$. Then

$$\text{dist}(\hat{c}_{t,i}, \hat{c}_{t+1,j}) \leq d$$

and

$$\text{dist}(\hat{c}_{t,i'}, \hat{c}_{t+1,j}) \leq d,$$

so

$$\text{dist}(\hat{c}_{t,i}, \hat{c}_{t,i'}) \leq 2d,$$

again contradicting the continuity condition, now in frame t . Thus π_t is injective. Since π_t is an injective map from a finite set of size k to itself, it is bijective, and hence $\pi_t \in S_k$.

Because the coordinates have constant bit-length and k is fixed, each predicate $\text{dist}(\hat{c}_{t,i}, \hat{c}_{t+1,j}) \leq d$ is computable by a constant-size circuit, and so is the selection of the unique j for each i . Hence, each π_t is computable by a constant-size circuit.

The induced global permutation is

$$\pi = \pi_{T-1} \circ \dots \circ \pi_0.$$

Since k is fixed, composing two permutations in S_k is computable by a constant-size circuit. Computing this product in a balanced binary tree gives depth $O(\log T) = O(\log n)$ and polynomial size.

Finally, the localization step is in NC^1 by assumption, and all per-frame sorting and matching operations are constant-size, repeated over $T = \Theta(n)$ frames. Thus the total size is polynomial, and the overall depth is $O(\log n)$. Therefore, the global permutation can be computed in NC^1 .

Since k is fixed, checking whether the resulting permutation π is the identity is computable by a constant-size circuit. Hence, $\text{TRACK}_k \in \text{NC}^1$. $\square$

Lemma 2. $\text{WORD}_{S_5} \leq_{\text{AC}^0} \text{TRACK}_k$ for any fixed $k \geq 5$.

Proof. We reduce from the word problem over S_5 with generators

$$\begin{aligned}\tau_1 &= (12), & \tau_2 &= (23), \\ \tau_3 &= (34), & \tau_4 &= (45),\end{aligned}$$

which is NC^1 -complete.

Fix constants $d > 0$, $L > 2d$, and $h > d$. Place the first five anchor positions at

$$p_i = ((i-1)L, 0), \quad i = 1, 2, 3, 4, 5,$$

so they lie on the same horizontal row and consecutive anchors are distance L apart. Place the remaining anchors

$$p_6, \dots, p_k$$

at fixed positions at distance greater than $2d$ from the region traversed by the gadget trajectories. Thus all inactive objects can remain stationary throughout the video without violating the continuity condition.

We use only the first five anchors to simulate the generators of S_5 ; the remaining anchors are occupied by stationary objects. The objects are rendered as fixed constant-size bright templates on a black background, so the object centers can be recovered trivially and the localization condition is satisfied.

For each $i \in \{1, 2, 3, 4\}$, we construct a constant-length gadget G_i that swaps the objects at p_i and p_{i+1} and leaves all other objects fixed:

$$p_i = ((i-1)L, 0), \quad p_{i+1} = (iL, 0).$$

The gadget moves the object at p_i along the three-segment path

$$\begin{aligned}((i-1)L, 0) &\rightarrow ((i-1)L, h) \\ &\rightarrow (iL, h) \rightarrow (iL, 0),\end{aligned}$$

and moves the object at p_{i+1} along the three-segment path

$$\begin{aligned}(iL, 0) &\rightarrow (iL, -h) \\ &\rightarrow ((i-1)L, -h) \rightarrow ((i-1)L, 0).\end{aligned}$$

All other objects remain fixed.

These two moving objects travel on different horizontal rows, so their trajectories remain disjoint. Since the active rows are $y = h$ and $y = -h$, and since the other anchors lie on $y = 0$ and are spaced by $L > 2d$, choosing h and the inactive anchors

as above ensures that at every moment any two distinct object centers are more than $2d$ apart. By subdividing each segment into a constant number of sufficiently short steps, we obtain a constant-length video in which every object moves by at most d between consecutive frames. Hence, each gadget satisfies the continuity condition and realizes exactly the adjacent transposition τ_i on the first five objects, while acting as the identity on objects $6, \dots, k$.

Now let

$$w = \tau_{a_1} \tau_{a_2} \cdots \tau_{a_m}$$

be an input word over $\{\tau_1, \tau_2, \tau_3, \tau_4\}$. We use the standard right-to-left composition convention, so the group element represented by w is

$$[w] = \tau_{a_m} \circ \cdots \circ \tau_{a_1},$$

that is, τ_{a_1} acts first, then τ_{a_2} , and so on. We map w to a video $V(w)$ by concatenating the gadgets

$$G_{a_1}, G_{a_2}, \dots, G_{a_m},$$

starting from the frame in which the k objects occupy the anchors $p_1, \dots, p_k$.

Each gadget ends with every object exactly at an anchor position, so the next gadget starts from the correct configuration. Therefore, the global permutation induced by the full video is exactly

$$[w] = \tau_{a_m} \circ \cdots \circ \tau_{a_1}$$

on the first five objects, and the identity on the remaining objects. Hence, the global permutation π is the identity on all k objects if and only if the word w evaluates to the identity in S_5 .

Finally, the reduction is a constant-block substitution: each input symbol τ_i is replaced by the constant-length gadget G_i . Hence, the output video is computable from the input word by a uniform AC^0 reduction. $\square$

Theorem 1. For any fixed $k \geq 5$, TRACK_k is NC^1 -complete.

Proof. Lemma 1 gives $\text{TRACK}_k \in \text{NC}^1$, and Lemma 2 gives NC^1 -hardness for every fixed $k \geq 5$ via an AC^0 reduction from WORD_{S_5} , which is NC^1 -complete. Therefore, TRACK_k is NC^1 -complete. $\square$

Model	Temporal Sampling			
	1×	2×	4×	8×
<i>Random</i>	0.33	0.33	0.33	0.33
Gemini 3 Pro	0.36	0.38	0.31	0.33
Gemini 3 Flash	0.34	0.36	0.33	0.31
Qwen3-VL-8B-Instruct	0.30	0.33	0.35	0.34
Qwen3-VL-8B-Thinking	0.32	0.31	0.34	0.31

Table 2: Accuracy under different temporal sampling. For models exposing a sampling rate (Qwen3-VL), the columns correspond to 1, 2, 4, and 8 FPS; for models with a fixed default rate (Gemini 3), they correspond to proportionally lengthened swaps, which scales the frames observed per swap equivalently.

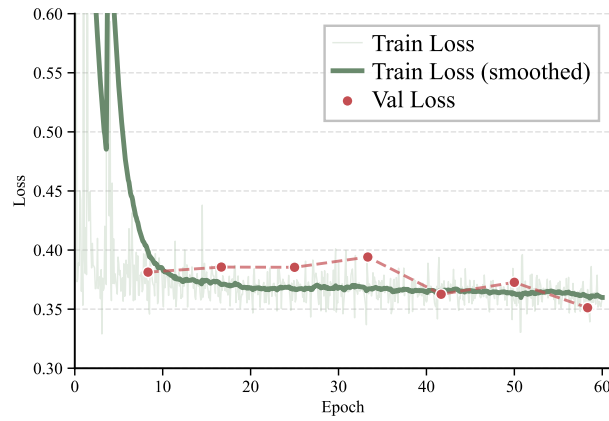


Figure 9: Training and validation loss for direct-answer training on 500 synthetic VET-Bench cups-game videos.

Gemini 3 Pro (Success Case)

Which cup contains the ball at the end of the video?

- (A) Left
- (B) Middle
- (C) Right

Output your final answer (A, B, or C) in the format: < option >.

To determine which cup contains the ball at the end of the video, we can track the position of the cup with the ball through each swap:

1. **Initial Position (00:00 - 00:01):** The video starts by lifting the **Left** cup, revealing the red ball underneath. The cup is then placed back down. **Ball Position:** Left
2. **First Swap (00:02 - 00:03):** The cup in the Left position swaps with the cup in the Middle position. The ball moves from Left → **Middle**.
3. **Second Swap (00:03 - 00:05):** The cup in the Middle position swaps with the cup in the Right position. The ball moves from Middle → **Right**.
4. **Third Swap (00:05 - 00:07):** The cup in the Right position swaps with the cup in the Left position (a long swap across the table). The ball moves from Right → **Left**.
5. **Fourth Swap (00:07 - 00:09):** The cup in the Left position swaps with the cup in the Middle position. The ball moves from Left → **Middle**.
6. **Fifth Swap (00:09 - 00:11):** The cup in the Middle position swaps with the cup in the Right position. The ball moves from Middle → **Right**.

At the end of the video, the cup containing the ball is in the **Right** position. C

Gemini 3 Pro (Failure Case)

Which cup contains the ball at the end of the video?

- (A) Left
- (B) Middle
- (C) Right

Output your final answer (A, B, or C) in the format: < option >.

To find the location of the ball, let's track the position of the cup containing the ball through each step of the video.

1. **Start (00:00 - 00:01):** The ball is shown to be under the **Left** cup.
2. **First Swap (00:02 - 00:03):** The cup on the Left (containing the ball) swaps places with the cup in the Middle. **Ball Position:** Middle.
3. **Second Swap (00:04 - 00:05):** The cup in the Middle (containing the ball) swaps places with the cup on the Right. **Ball Position:** Right.
4. **Third Swap (00:06 - 00:07):** The cup on the Right (containing the ball) swaps places with the cup in the Middle. **Ball Position:** Middle.
5. **Fourth Swap (00:08 - 00:09):** The cup in the Middle (containing the ball) swaps places with the cup on the Left. **Ball Position:** Left.
6. **Fifth Swap (00:10 - 00:11):** The cup on the Left (containing the ball) swaps places with the cup in the Middle. **Ball Position:** Middle.

At the end of the video, the ball is under the cup in the middle position. B

Figure 10: Examples of Gemini 3 Pro responses on the VET-Bench cups-game task. (Top) A successful case where the model correctly tracks all the intermediate states. (Bottom) A failure case where perception errors (highlighted in red) lead to an incorrect final prediction.

Gemini 3 Pro (Success Case)

Where is the Queen of Hearts at the end of the video?

- (A) Left
- (B) Middle
- (C) Right

Output your final answer (A, B, or C) in the format: < option >.

To determine the final position of the Queen of Hearts, let's track its movement step-by-step through the video:

1. **Start (00:00):** The Queen of Hearts is revealed in the **Middle**. It then flips face down.
2. **First Move (00:01 - 00:03):** The card in the Middle (Queen) and the card on the Left swap places. The Middle card moves over to the Left position, and the Left card moves under to the Middle position. Current Position of Queen: **Left**.
3. **Second Move (00:03 - 00:05):** The card now in the Middle (which was the empty card) and the card on the Left (Queen) swap places again. The Middle card moves over to the Left, and the Left card moves under to the Middle. Current Position of Queen: **Middle**.
4. **Third Move (00:05 - 00:07):** The card on the Left (empty) and the card in the Middle (Queen) swap. The Left card moves over to the Middle, and the Middle card moves under to the Left. Current Position of Queen: **Left**.
5. **Fourth Move (00:07 - 00:09):** The card on the Left (Queen) and the card in the Middle (empty) swap. The Left card moves over to the Middle, and the Middle card moves under to the Left. Current Position of Queen: **Middle**.
6. **Fifth Move (00:09 - 00:11):** The card in the Middle (Queen) and the card on the Right (empty) swap. The Right card moves over to the Middle, and the Middle card moves under to the Right. Current Position of Queen: **Right**.

At the end of the video, the Queen of Hearts is in the Right position. C

Gemini 3 Pro (Failure Case)

Where is the Queen of Hearts at the end of the video?

- (A) Left
- (B) Middle
- (C) Right

Output your final answer (A, B, or C) in the format: < option >.

Based on the visual evidence provided in the frames, here is the step-by-step tracking of the Queen of Hearts:

1. **Start (00:00 - 00:01):** The Queen of Hearts is the face-up card on the **Right**. It flips face-down at 00:01. **Current Position:** Right
2. **Swap 1 (00:01 - 00:03):** The first movement involves the Left and Middle cards. The Left card moves high/right, and the Middle card moves low/left. The Right card (Queen) remains stationary. **Current Position:** Right
3. **Swap 2 (00:03 - 00:05):** The second movement involves the Middle and Right cards. The Right card (Queen) moves high/left to the middle position. The Middle card moves low/right to the right position. **Current Position:** Middle
4. **Swap 3 (00:05 - 00:07):** The third movement again involves the Middle and Right cards. The Middle card (Queen) moves low/right back to the right position. The Right card moves high/left to the middle position. **Current Position:** Right
5. **Swap 4 (00:07 - 00:09):** The fourth movement involves the Left and Middle cards. The Right card (Queen) remains stationary. **Current Position:** Right
6. **Swap 5 (00:09 - 00:11):** The final movement involves the Left and Middle cards again. The Right card (Queen) remains stationary. **Final Position:** Right

Therefore, at the end of the video, the Queen of Hearts is in the Right position. C

Figure 11: Examples of Gemini 3 Pro responses on the VET-Bench cards-game task. (Top) A successful case where the model correctly tracks all the intermediate states. (Bottom) A failure case where perception errors (highlighted in red) lead to an incorrect final prediction.

Qwen3-VL-8B-Instruct

Which cup contains the ball at the end of the video?

- (A) Left
- (B) Middle
- (C) Right

Output your final answer (A, B, or C) in the format: .

At the end of the video, all three cups are back in their original positions, and there is no visible ball. However, based on the sequence:

- At 0:02, the ball appears under the rightmost cup.
- At 0:04, the rightmost cup moves slightly but still covers the ball.
- From 0:06 onwards, the cups shuffle around, but by 0:12, they return to their initial positions.

Since the ball was hidden under the rightmost cup at 0:02 and **no one moved it afterward (the cups just shuffled)**, it should still be under the rightmost cup when they return to their positions. Therefore, the ball is under the right cup at the end.

Kimi K2.5

Which cup contains the ball at the end of the video?

- (A) Left
- (B) Middle
- (C) Right

Output your final answer (A, B, or C) in the format: .

Looking at the video sequence:

1. At [00:00:00.500], the middle cup is lifted, revealing a red ball underneath.
2. At [00:00:01.500], the middle cup is placed back down, covering the ball.
3. **The cups then move around in various patterns.**
4. At [00:00:12.100], the cups return to their original positions (left, middle, right).

Since the ball was placed under the middle cup at [00:00:01.500], and that cup is back in the center position at the end of the video, **the ball remains under the middle cup.**

Doubao-Seed-1.8

Which cup contains the ball at the end of the video?

- (A) Left
- (B) Middle
- (C) Right

Output your final answer (A, B, or C) in the format: .

To determine which cup contains the ball, observe the video sequence:

1. At the start, the right cup is lifted, revealing the red ball underneath (timestamps ~0.2–0.8 seconds).
2. After the right cup is placed back, the cups move around, **but the ball remains under the right cup through the movements** (as the initial placement and subsequent cup movements confirm the ball's position relative to the right cup).

Thus, the cup containing the ball at the end is the right one.

Figure 12: Examples of model responses on VET-Bench. Errors are highlighted in red.

Gemini 3 Flash (Success Case)

Which cup contains the ball at the end of the video?

- (A) Left
- (B) Middle
- (C) Right

Output your final answer (A, B, or C) in the format: .

The video begins with three cups and a red ball. The red ball is positioned on the right side. At the start of the video, the right cup is lowered to cover the ball. No shuffling or additional movements of the cups take place throughout the rest of the video. Therefore, the ball remains under the right cup.

Gemini 3 Flash (Hallucination)

Which cup contains the ball at the end of the video?

- (A) Left
- (B) Middle
- (C) Right

Output your final answer (A, B, or C) in the format: .

To find the location of the ball, we need to carefully track the movements of the cups throughout the video:

1. **Start (00:00 - 00:01):** The ball is placed under the **Right** cup.
2. **Swap 1 (00:02):** The **Middle** cup and the **Right** cup swap positions. The ball is now in the **Middle** position.
3. **Swap 2 (00:03):** The **Left** cup and the **Middle** cup swap positions. The ball is now in the **Left** position.
4. **Swap 3 (00:04):** The **Middle** cup and the **Right** cup swap positions. The ball remains in the **Left** position.
5. **Swap 4 (00:05):** The **Left** cup and the **Middle** cup swap positions. The ball is now in the **Middle** position.
6. **Swap 5 (00:06):** The **Middle** cup and the **Right** cup swap positions. The ball is now in the **Right** position.
7. **Swap 6 (00:07):** The **Left** cup and the **Middle** cup swap positions. The ball remains in the **Right** position.
8. **Swap 7 (00:08):** The **Middle** cup and the **Right** cup swap positions. The ball is now in the **Middle** position.

At the end of the video, the ball is under the cup in the middle position.

Figure 13: A success case and a hallucination case by Gemini 3 Flash on zero-swap videos. Gemini 3 models often correctly identify the initial state but then hallucinate a long sequence of non-existent swaps (as shown in red), causing the final prediction to be incorrect.

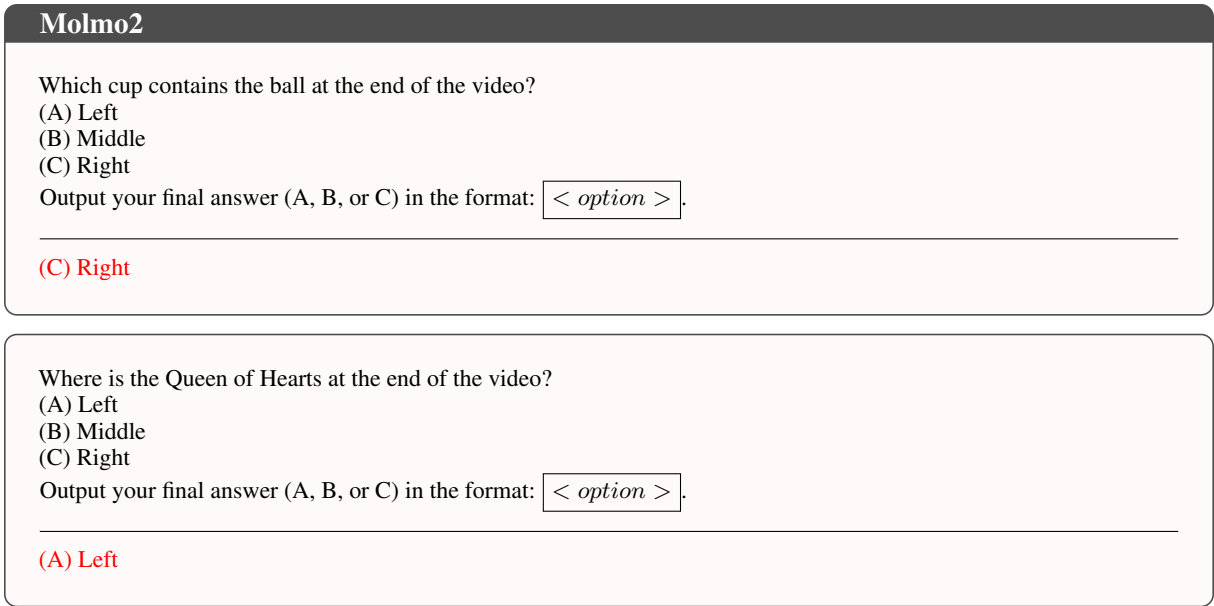


Figure 14: Examples of Molmo2’s prompts and responses on VET-Bench. Molmo2 responds in a direct-answer mode without explicitly performing spatiotemporal tracking, resulting in near-random performance.

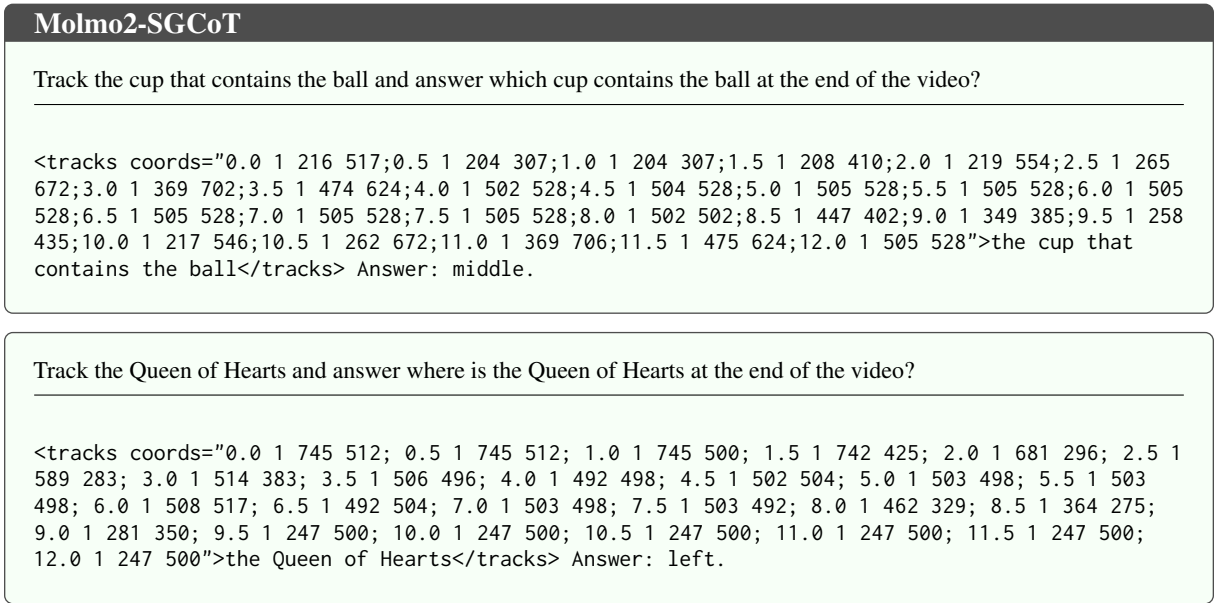


Figure 15: Examples of Molmo2-SGCoT’s prompts and responses on VET-Bench. Top: cups game; Bottom: cards game.

Molmo2-SGCoT (Failure Cases)

Track the cup that contains the ball and answer which cup contains the ball at the end of the video?

<tracks coords="0.0 1 774 517;0.5 1 804 307;1.0 1 804 307;1.5 1 794 402;2.0 1 765 480;2.5 1 647 333;3.0 1 458 289;**3.5 1 292 378;4.0 1 765 480**;4.5 1 647 333;5.0 1 458 289;5.5 1 292 378;6.0 1 228 491;6.5 1 349 333;7.0 1 542 289;7.5 1 713 374;8.0 1 774 498;8.5 1 707 383;9.0 1 607 367;9.5 1 528 422;10.0 1 504 498;10.5 1 553 389;11.0 1 647 367;11.5 1 742 422;12.0 1 774 517">the cup that contains the ball</tracks> Answer: right.

Track the Queen of Hearts and answer where is the Queen of Hearts at the end of the video?

<tracks coords="0.0 1 750 517;0.5 1 750 517;**1.0 1 745 500;4.5 1 442 775**;5.0 1 286 646;5.5 1 252 538;6.0 1 355 738;6.5 1 539 788;7.0 1 705 662;7.5 1 750 500;8.0 1 750 500;8.5 1 750 500;9.0 1 750 500;9.5 1 750 500;10.0 1 714 338;10.5 1 622 281;11.0 1 536 342;11.5 1 500 496;12.0 1 500 496">the Queen of Hearts</tracks> Answer: middle.

Figure 16: Examples of Molmo2-SGCoT failure cases. Errors typically arise in SGCoT when the model fails to differentiate between visually identical objects. This manifests as abrupt “jumps” in the spatial (top) or temporal (bottom) domains (highlighted in red), leading to incorrect terminal locations and subsequent failures in the final answer.

Model	Direct Prompting	SGCoT Prompting
<i>Random</i>	0.333	0.333
Gemini 3 Pro	0.364	0.356
Gemini 3 Flash	0.336	0.334
Qwen3-VL-8B-Thinking	0.310	0.310
Qwen3-VL-8B-Instruct	0.336	0.334
Doubao-Seed-1.8	0.306	0.312
GLM-4.6V-Flash	0.290	0.290
Kimi K2.5	0.294	0.306
Molmo2-8B	0.320	0.312
Molmo2-SGCoT (fine-tuned)	—	0.914

Table 3: Accuracy under direct prompting and SGCoT-style prompting.

Hyperparameter	Value
Base Model	Molmo2-8B
Training Method	QLoRA
Gradient Accumulation Steps	4
Batch Size	64
Learning Rate	10^{-4}
Training Samples	300
Number of Epochs	1
LoRA Configuration	
LoRA Rank (r)	16
LoRA Alpha (α)	16
Target Modules	att_proj, attn_out, ff_proj, ff_out

Table 4: Molmo2-SGCoT training details. We apply QLoRA (Dettmers et al., 2023) to fine-tune the language model while keeping the vision encoder weights frozen.