TOC-BENCH: DIAGNOSING TEMPORAL OBJECT CONSISTENCY IN VIDEO LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Video large language models (Video-LLMs) have made substantial progress in general video understanding, yet their ability to maintain *temporal object consistency* remains insufficiently evaluated. This capability requires tracking the identity, state, and persistence of the same object through occlusion, disappearance, reappearance, and interactions with other objects. We introduce *TOC-Bench*, a diagnostic benchmark that grounds temporal questions in explicit object tracks and structured event timelines. Answers to candidate questions are determined from object-event records before natural-language generation. A three-layer *temporal-necessity filtering* protocol probes text-only, single-frame, and frame-shuffled shortcuts, removing 60.7% of 45,527 candidates. Expert verification then yields 2,323 QA pairs over 1,951 real videos, spanning 10 diagnostic dimensions and four task formats with deterministic scoring. Experiments on 26 Video-LLMs reveal substantial difficulties in repeated-event counting, exact event ordering, and detecting unsupported premises about objects or events: the best-performing model achieves 47.1% accuracy, compared with a three-person human average of 89.3%. Controlled 16/32/64-frame comparisons show that increasing the number of uniformly sampled frames alone does not consistently improve performance. TOC-Bench offers new instance-linked annotations and a focused platform for diagnosing and improving object-aware temporal reasoning. The resource is available at `https://anonymous.4open.science/r/toc_bench-1355`.

1 INTRODUCTION

Video-LLMs have made rapid progress in video QA, event description, and temporal reasoning through video-instruction tuning (Maaz et al., 2024; Lin et al., 2024; Li et al., 2025b; Cheng et al., 2024; Zhang et al., 2024a) and stronger multimodal systems (Wang et al., 2024a; Chen et al., 2024; Wang et al., 2025b; Ryoo et al., 2024; Bai et al., 2025b). Evaluation has expanded from short-video captioning and QA (Xu et al., 2016; 2017; Yu et al., 2019) to broad temporal reasoning, long-video comprehension, and shortcut-aware benchmarks (Li et al., 2024b; Fu et al., 2025; Fang et al., 2024; Wu et al., 2024; Cores et al., 2024; Fu et al., 2026). These advances raise a finer-grained question: can Video-LLMs maintain object-level consistency?

We refer to this capability as *temporal object consistency* (TOC): the ability to track the identity, state, and persistence of the same object throughout a video. This capability is important for real-world video understanding, where objects may be hidden, leave the scene, reappear, or interact over time. For instance, distinguishing whether an object disappears because it is occluded or leaves the frame, identifying whether a reappearing object is the same instance, and determining whether one object's state change precedes an event involving another object all require temporally ordered evidence anchored to specific objects. Without this capability, a model may correctly recognize individual events but still fail to reason about object persistence and object-level temporal relations.

Temporal object consistency remains insufficiently evaluated as a unified capability. Existing benchmarks already test counting, ordering, and object interactions (Jang et al., 2017; Li et al., 2024b; Cores et al., 2024). For TOC, the central requirement is to bind these operations to the history of a specific object instance: which object returns, how often it returns, and what happens to another object at that time. Recent shortcut-aware benchmarks also show that some video questions can be solved from static frames or question-answer text alone (Cores et al., 2024; Liu et al., 2024; Krojer

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

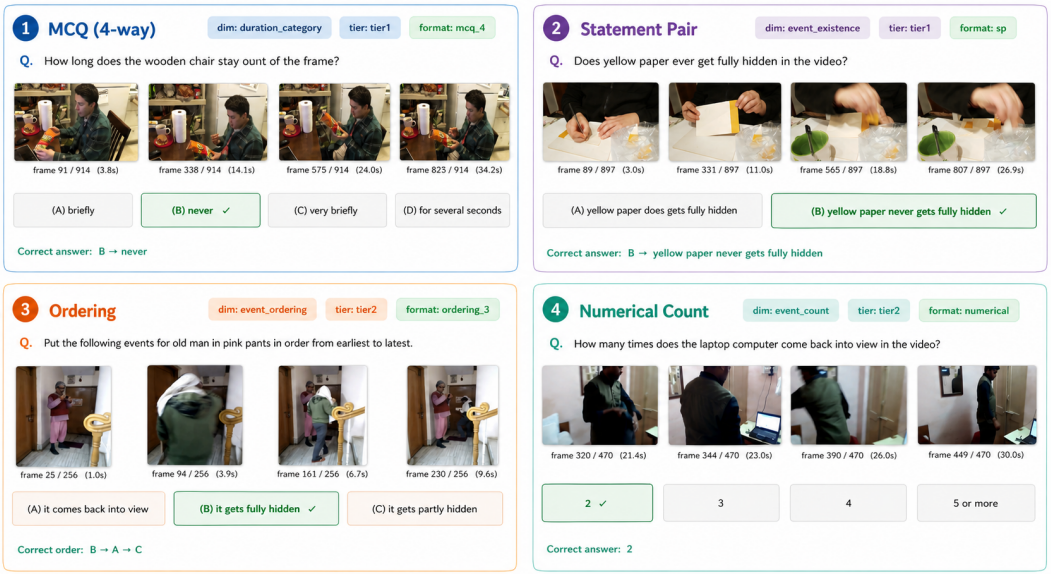

Figure 1: Representative TOC-Bench QA examples. The benchmark supports four task formats with deterministic scoring: four-way multiple choice, statement-pair judgment, event ordering, and numerical counting. Correct answers are highlighted in green.

et al., 2025). These observations motivate evaluation built around explicit object tracks and event timelines, with filtering to reduce shortcut-solvable questions.

To address this gap, we introduce TOC-Bench, a diagnostic benchmark for evaluating temporal object consistency in Video-LLMs. TOC-Bench is *grounded in object tracks*: base questions refer to tracked objects, with answers derived from structured object-event timelines before natural-language generation. It contains 2,323 human-verified QA pairs over 1,951 real videos, covers 10 diagnostic dimensions in three difficulty tiers, and supports four task formats with deterministic scoring: four-way multiple choice, statement pair, numerical count, and event ordering. Temporal-necessity filtering reduces shortcut-solvable questions, while hallucination-aware variants test whether models accept unsupported premises about objects or events. These variants complement recent efforts to evaluate hallucinations in video-language models (Li et al., 2025a; Wang et al., 2024c). Representative examples are shown in Fig. 1.

We evaluate 26 representative Video-LLMs on TOC-Bench and find that temporal object consistency remains a major challenge. Current models perform far below humans, especially on repeated-event counting, exact event ordering, and premise verification. These findings highlight the need to evaluate object-level temporal reasoning alongside general video understanding.

Our contributions are summarized as follows:

- We formulate **temporal object consistency** as an evaluation axis that binds temporal judgments to specific object instances and their histories, covering 10 diagnostic dimensions in three difficulty tiers.
- We instantiate this axis as **TOC-Bench**, a diagnostic benchmark grounded in object tracks. Base questions refer to tracked objects, with answers derived from structured object-event timelines. The benchmark contains 2,323 human-verified QA pairs over 1,951 videos and supports four task formats with deterministic scoring.
- We propose a **shortcut-aware and hallucination-aware diagnostic protocol**. Starting from 45,527 generated QA pairs, the temporal-necessity filter removes items showing shortcut success under text-only, single-frame, or frame-shuffled probes and retains 17,900 candidates. Hallucination-aware variants and hallucination diagnostic accuracy (HDA) assess whether models reject unsupported premises about objects or events.

2 Related Work

2.1 Video Large Language Models

Recent Video-LLMs extend vision-language models (VLMs) to video understanding through video-specific instruction tuning, long-context modeling, and large-scale multimodal training. Representative models such as VideoChatGPT (Maaz et al., 2024), Video-LLaVA (Lin et al., 2024), VideoChat (Li et al., 2025b), VideoLLaMA 2 (Cheng et al., 2024), LLaVA-NeXT-Video (Zhang et al., 2024a), and LLaVA-Video (Zhang et al., 2024b) adapt image-language models to video QA, captioning, and dialogue. More recent models, including VideoChat-Flash (Li et al., 2024d), InternVideo2/2.5 (Wang et al., 2024b; 2025b), Qwen2.5-VL (Bai et al., 2025b), Qwen3-VL (Bai et al., 2025a), InternVL3 (Zhu et al., 2025), and Tarsier2 (Yuan et al., 2025a), further improve long-context perception and general multimodal reasoning. Their diverse frame-sampling and visual-processing strategies make explicit reporting of input settings important when evaluating object-level temporal reasoning.

2.2 Video QA Benchmarks and Temporal Evaluation

Video QA benchmarks have evolved from datasets such as MSVD-QA and MSRVTT-QA (Xu et al., 2017), ActivityNet-QA (Yu et al., 2019), TGIF-QA (Jang et al., 2017), NExT-QA (Xiao et al., 2021), and STAR (Wu et al., 2021), to broader Video-LLM evaluation suites such as EgoSchema (Mangalam et al., 2023), MVBench (Li et al., 2024b), Video-MME (Fu et al., 2025), Video-MME-v2 (Fu et al., 2026), MMBench-Video (Fang et al., 2024), LongVideoBench (Wu et al., 2024), and LVBench (Wang et al., 2025a). These benchmarks cover activity understanding, long-form comprehension, multi-domain QA, and broad temporal reasoning. Counting and ordering are established operations; TOC-Bench contributes newly constructed questions whose referents and answers are linked to object-specific event histories.

Recent benchmarks also examine temporal evidence and shortcuts. TempCompass (Liu et al., 2024) and TVBench (Cores et al., 2024) control static-frame or text-based shortcuts, while Temporal-Bench (Cai et al., 2024) evaluates fine-grained temporal understanding and addresses answer-option bias. RTV-Bench (Xun et al., 2025) evaluates continuous perception and reasoning in real-time video. VITATECS (Li et al., 2024c) shares our diagnostic motivation through counterfactual descriptions of temporal concepts. TOC-Bench instead grounds base questions in per-instance tracks and event timelines. SVCBench (Liu et al., 2026) examines state maintenance through streaming counts; TOC-Bench complements it with full-video diagnostics of identity, visibility, conditional state, and false premises.

2.3 Object-Centric Video Grounding and Tracking

Object-centric video grounding has advanced through segmentation, tracking, and referring video object segmentation methods such as Segment Anything (Kirillov et al., 2023), SAM2 (Ravi et al., 2024), Track Anything (Yang et al., 2023), SAM2Long (Ding et al., 2025), and VideoRefer Suite (Yuan et al., 2025b). These methods provide tools for localizing, segmenting, and following objects in videos. TOC-Bench uses tracking for benchmark construction rather than as an evaluation target: it asks whether Video-LLMs can recover the relevant object history from video to answer temporally grounded questions. Construction tracks and event labels are not provided to evaluated models.

3 TOC-Bench: Temporal Object Consistency Benchmark

TOC-Bench evaluates whether Video-LLMs can track object identity, state, and persistence over time. We use *diagnostic* to mean that the benchmark targets this capability through structured QA items with automatic scoring, rather than measuring general video QA performance.

TOC-Bench contains 2,323 human-verified QA pairs from 1,951 unique videos across four public sources: Charades (Sigurdsson et al., 2016), Perception Test (Patraucean et al., 2023), OVIS (Qi et al., 2022), and MOSE (Ding et al., 2023). The total video duration is 15 h 37 min, with a mean of 28.8 s. Questions are organized around object grounding (O), temporal evidence (T), and consistency constraints (C), forming three difficulty tiers: Tier 1 (O+T) for single-object temporal reasoning,

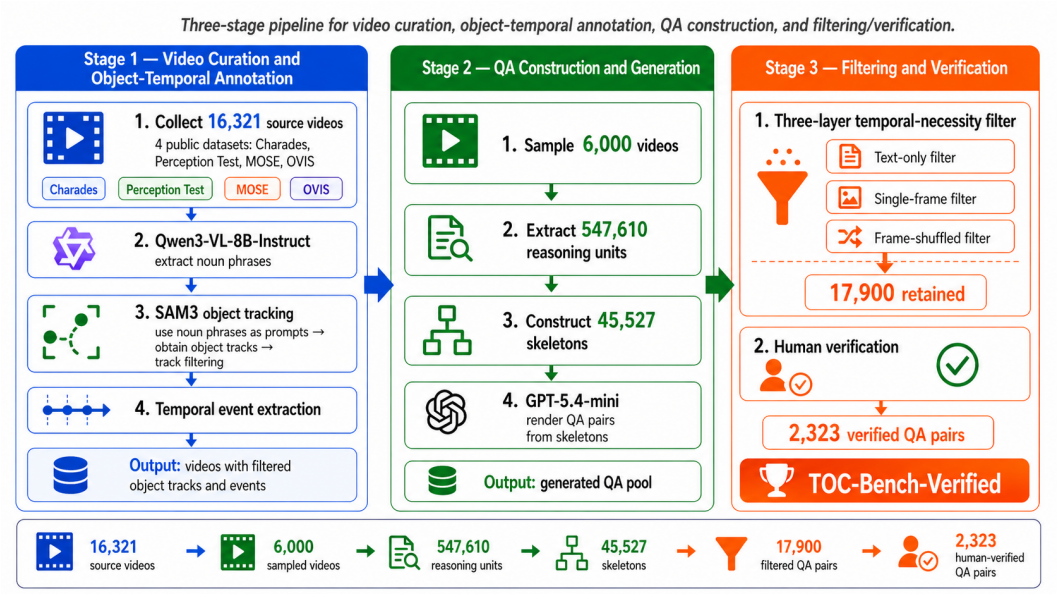

Figure 2: The construction pipeline of TOC-Bench. Of 17,900 filtered candidates, 3,000 are sampled for expert verification and 2,323 are retained.

Tier 2 (O+T+C) for cross-time consistency of one object, and Tier 3 (O+T+C×2) for consistency reasoning across two objects. TOC-Bench covers 10 diagnostic dimensions and four task formats: multiple choice, statement pair, numerical count, and event ordering. Together, they probe object persistence, event timing, state change, repeated-event accumulation, identity preservation, spatial change, and cross-object temporal relations.

3.1 Benchmark Construction

TOC-Bench is built through a three-stage pipeline, as shown in Fig. 2. Stage 1 derives object tracks and event timelines from source videos. Stage 2 converts these structured records into reasoning units and candidate QA pairs. Stage 3 applies temporal-necessity filtering and human verification to obtain the final benchmark.

Video curation and object-track annotation. As shown in Fig. 3, TOC-Bench starts from 16,321 candidate videos from four public datasets. Charades provides daily indoor activities with rich object interactions, while Perception Test contains scripted videos designed to assess physical reasoning, memory, and temporal reasoning. OVIS and MOSE contribute occlusion-heavy videos for constructing reappearance, occlusion, and cross-object reasoning questions. Together, these sources provide real-video observations of the object-level temporal phenomena underlying the newly constructed QA items.

For each video, we use Qwen3-VL-8B-Instruct (Bai et al., 2025a) with up to 10 sampled frames to propose distinct physical objects and people. The extracted noun phrases are used as prompts for SAM3 (Carion et al., 2025) to obtain candidate per-frame object tracks. We remove low-quality, fragmented, or visually ambiguous tracks, then derive appearance, disappearance, occlusion, and reappearance events, as well as cross-object relations, from the retained tracks. Qwen3-VL-8B-Instruct and SAM3 support scalable candidate annotation, while downstream QA construction relies on filtered tracks and rule-derived event timelines. Sampling and annotation details are provided in Appendix A.

QA construction and generation. The QA pairs are newly constructed from object tracks and event timelines rather than inherited from the source datasets. We sample 6,000 videos, prioritizing valid tracks and clear temporal changes, then construct 547,610 reasoning units specifying the subject, event, dimension, answer type, candidate answer, and admissible distractors. For the apple example in Fig. 19, the same object's timeline contains four reappearance events. The unit specifies the apple

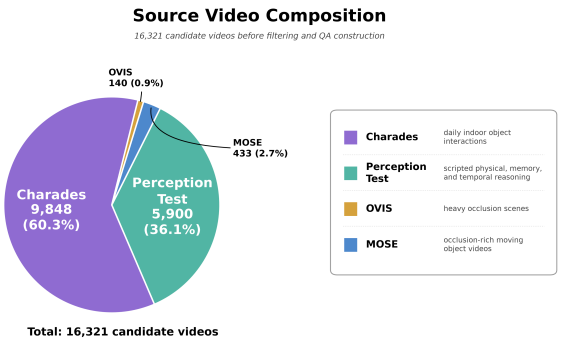

Figure 3: Candidate source-video composition before filtering ($N = 16{,}321$). Final video and QA counts are reported separately in Section 3.2.

as the subject, `reappear` as the event type, and 4 as the count before question wording is generated; the final verified answer remains 4. Thus, the count describes one object's event history, not the number of objects visible in a frame.

We then convert reasoning units into question skeletons, which specify the question structure, answer options, correct answer, and task format. At this stage, we also introduce hallucination-aware variants with nonexistent subjects or events, together with visually unsupported distractor options. This produces 45,527 skeletons spanning 10 diagnostic dimensions and four task formats. GPT-5.4-mini performs surface realization, converting each skeleton into fluent text while preserving its object references, answer labels, temporal buckets, numerical answers, and ordering relations. Implementation details are provided in Appendix B.

Filtering and verification. After QA generation, we apply a three-layer temporal-necessity filter. The text-only and single-frame layers remove items that meet the shortcut-success criteria under text-only and individual-frame probes, respectively. The frame-shuffled layer removes items answered correctly with both ordered and shuffled frame sequences. Together, these layers screen for shortcuts based on language priors, static visual cues, and unordered frame evidence. The filtering process retains 17,900 of 45,527 generated QA pairs as candidates for human verification.

We sample 3,000 filtered items by dimension for human verification. Five experts each inspect a disjoint set of 600 video–question–options–answer units. Minor issues such as unclear wording, duplicated options, or slight reference ambiguity are manually revised when the intended evidence and answer remain valid. Items are removed only when they have severe evidence or label problems, such as tracking errors, insufficient visual support, incorrect answers, or a clear mismatch between the QA item and the video. This process retains 2,323 QA pairs and rejects 677 (22.6%).

Upstream models support candidate construction and wording, while structured object-event records determine candidate answers. Rule-based checks and human verification establish which items are retained. Additional details on filtering and construction-model dependence are provided in Appendices C and D.

3.2 Dataset Statistics

TOC-Bench prioritizes diagnostic validity and human verification over raw scale. Expert review of 3,000 items sampled from the 17,900 filtered candidates yields 2,323 QA pairs over 1,951 videos, averaging 1.19 QA pairs per video. The final videos comprise 1,149 from Charades, 712 from Perception Test, 76 from MOSE, and 14 from OVIS.

As shown in Fig. 4, temporal location, duration category, event count, and conditional state are the largest diagnostic dimensions. TOC-Bench uses structured answer formats with deterministic scoring to support reproducible evaluation of object-level temporal consistency. It includes 1,355 multiple-choice questions, 563 statement-pair questions, 300 numerical questions, and 105 event-ordering questions. The three tiers contain 1,296, 562, and 465 items, respectively; thus, 44.2% of the dataset targets cross-time consistency for one object or consistency reasoning across two objects.

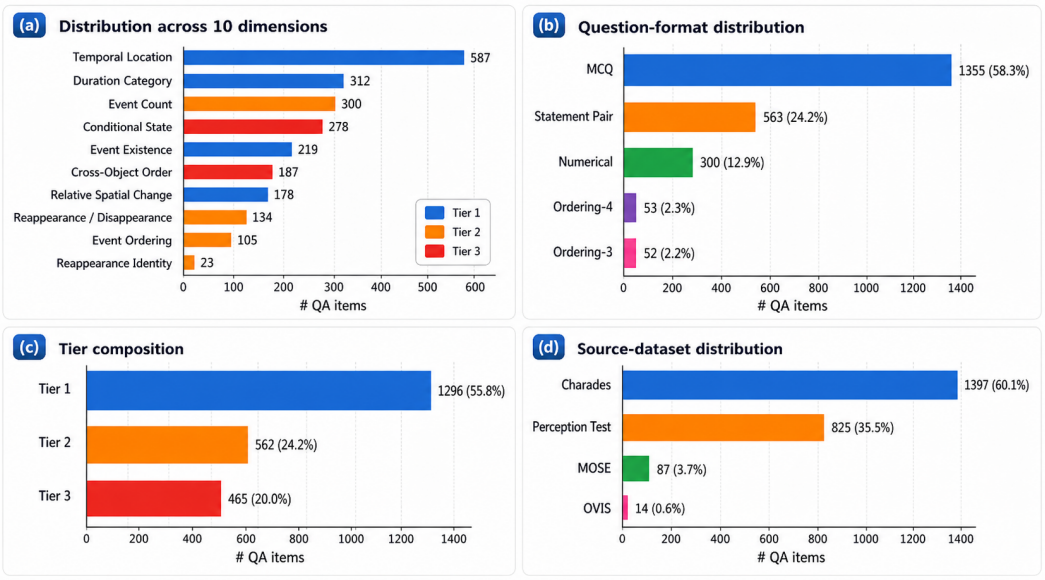

Figure 4: Overall composition of TOC-Bench. (a)-(d) depict the distribution of QA items across the 10 diagnostic dimensions, with colors indicating tier assignments; the question-format distribution; the tier-level composition; and the source-dataset distribution, respectively.

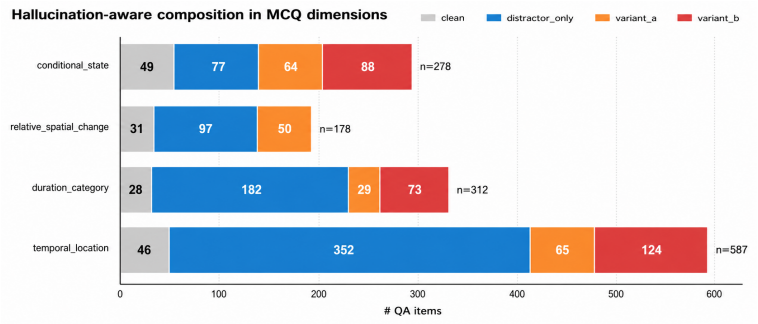

Figure 5: Hallucination-aware composition of TOC-Bench in multiple-choice dimensions. The stacked bars show the number of clean items, distractor-only items, nonexistent-subject variants (variant A), and nonexistent-event variants (variant B) in each dimension.

TOC-Bench includes hallucination-aware items to test whether models can reject plausible but visually unsupported premises. As shown in Fig. 5, these items occur in four multiple-choice dimensions: temporal location, duration category, relative spatial change, and conditional state. In addition to clean questions, the benchmark includes distractor-only items and two false-premise variants: Variant A contains 208 items whose queried subjects do not exist in the video, and Variant B contains 285 items whose subjects exist but whose queried events do not occur. Both subsets are part of the 2,323-item benchmark.

TOC-Bench complements the benchmarks in Table 1 by organizing evaluation around object histories. TGIF-QA (Jang et al., 2017) evaluates counting and temporal transitions, while SVCBench (Liu et al., 2026) tracks numerical state at successive streaming queries. TOC-Bench uses a common representation of object tracks and events to derive questions about visibility, identity, chronology, and conditional state, together with false-premise variants. This links complementary judgments to instance-specific temporal evidence, providing new annotations and diagnostic coverage through automatically scored QA. Additional dataset analyses appear in Appendix E.

Table 1: Comparison between TOC-Bench and representative video understanding benchmarks. TOC-Bench is designed as a diagnostic benchmark for temporal object consistency. Object-track grounding denotes a common track/event-based QA construction, not the mere presence of object-related questions. "Temporal filter" indicates whether the benchmark construction or evaluation explicitly controls text-only, single-frame, or frame-order shortcuts.

Benchmark	Videos	QA pairs	Format	Object-track grounded	Temporal filter	Halluc.-aware	Auto. grading
MVBench	4,000	4,000	MCQ	No	No	No	Yes
TVBench	–	2,654	MCQ	No	Yes	No	Yes
EgoSchema	5,000+	5,000+	MCQ	No	Partial	No	Yes
VITATECS	–	–	Text-video matching	No	Yes	No	Yes
Video-MME	900	2,700	MCQ	No	No	No	Yes
Video-MME-v2	800	3,200	MCQ(8-option)	No	Partial	Partial	Yes
LongVideoBench	3,763	6,678	MCQ	No	Partial	No	Yes
TOC-Bench	1,951	2,323	MCQ/SP Num/Ord	Yes	Yes	Yes	Yes

4 Experiments

4.1 Evaluation Setup

We evaluate 26 Video-LLMs, including closed-source multimodal systems, standard open-source models, and open-source thinking or reasoning variants. Input sampling and preprocessing follow the corresponding model documentation; Table 2 reports the frame count or sampling rate used in each run. Most models use 32 uniformly sampled frames. TOC-Bench specifies the task instructions, output formats, answer extraction, and scoring. Frame-budget comparisons hold all other settings fixed within each model. Further protocol details appear in Appendix F.

Closed-source models are accessed through official APIs; local inference uses two A800 GPUs with 80GB memory each. Models receive video evidence and QA instructions, without construction tracks or event timelines. The human baseline is separate from construction-stage verification: it averages the accuracies of three solvers on the final videos and questions, also without construction metadata.

Overall denotes micro-averaged accuracy over all 2,323 QA items; parsing and evaluation failures count as incorrect. Scoring is deterministic for all formats. Multiple-choice and statement-pair questions require an exact label match; numerical predictions are normalized to the controlled answer set and compared with the reference answer; ordering questions require an exact sequence match. Per-dimension scores are interpreted with their format-specific answer spaces and baselines. In addition to Overall and per-dimension accuracy, hallucination diagnostic accuracy (HDA) measures accuracy on false-premise questions, complementing performance on valid-premise items. Let $\mathcal{H}_A$ and $\mathcal{H}_B$ denote the nonexistent-subject and nonexistent-event subsets, respectively:

$$\text{HDA} = \frac{\sum_{q \in \mathcal{H}_A \cup \mathcal{H}_B} \mathbf{1}[\hat{a}_q = a_q]}{|\mathcal{H}_A \cup \mathcal{H}_B|}, \tag{1}$$

where $\hat{a}_q$ and a_q denote predicted and correct answers. HDA-S and HDA-E report accuracy on the 208 nonexistent-subject and 285 nonexistent-event items, respectively. For six representative models, 20,000 QA-item bootstrap resamples yield 95% percentile confidence intervals. ReID contains 23 items and contributes 0.99% of the weight in Overall. Detailed uncertainty estimates and answer-balance analyses appear in Appendix G.5.

4.2 Main Results

TOC-Bench reveals a large human–model gap. As shown in Table 2 and Fig. 6, the three-person human average is 89.3%, while the best-performing model, GPT-5.5, reaches 47.1%. GPT-5.5's 95% item-bootstrap interval is [45.1, 49.1]. The gap highlights substantial room to improve reasoning about object histories.

Table 2: Main evaluation results on TOC-Bench. All values are percentages. Frames denotes the sampled frame count unless fps is specified. Random denotes the expected accuracy of uniform guessing within the valid answer space of each item. Exist = event existence, Count = event count, ReID = reappearance identity, R/D = reappearance / disappearance, X-Order = cross-object order, Order = event ordering, T-Loc = temporal location, Dur = duration category, Cond = conditional state, and Move = relative spatial change. HDA denotes hallucination diagnostic accuracy, defined in Eq. (1).

Model	Frames	Overall	Exist	Count	ReID	R/D	X-Order	Order	T-Loc	Dur	Cond	Move	HDA
Human	-	89.3	94.5	84.0	100.0	75.4	87.2	80.0	94.5	89.7	91.4	87.6	89.5
Random	-	30.4	50.0	25.0	50.0	50.0	50.0	10.4	25.0	25.0	25.0	25.0	25.0
Closed-source models													
GPT-5.5 (OpenAI, 2026)	32	47.1	68.9	18.3	91.3	78.4	63.1	24.8	48.6	44.6	41.7	43.8	42.8
Kimi-K2.6 (Moonshot AI, 2026)	32	45.0	56.6	19.3	87.0	69.4	63.6	22.9	47.4	30.1	56.5	43.8	52.9
Gemini-3.1-Pro-Preview (Google, 2026)	32	43.9	56.2	18.7	87.0	67.2	57.2	17.1	45.0	40.1	50.0	43.8	47.7
Grok-4.3 (xAI, 2026)	32	43.8	48.9	19.7	95.7	70.9	61.0	17.1	42.9	39.7	51.8	46.6	53.8
Seed-2.0-Lite (ByteDance Seed Team, 2026)	32	42.0	57.5	24.3	95.7	49.3	52.9	21.9	40.9	43.9	43.5	38.2	41.2
GLM-5V-Turbo (Z.AI, 2026)	32	37.9	52.1	13.7	60.9	67.9	56.1	17.1	34.8	42.3	36.7	33.1	34.7
Gemini-3.1-Flash-Lite-Preview	32	33.8	37.9	6.0	91.3	58.2	48.1	16.2	38.0	30.1	39.6	28.7	41.2
GPT-5.4-mini	32	33.1	46.1	21.3	69.6	47.8	49.2	11.4	34.1	21.8	36.3	28.7	28.4
Mimo-V2-Omni (Xiaomi, 2026)	32	31.6	47.5	4.7	69.6	57.5	47.6	7.6	30.3	26.9	42.1	27.0	35.1
Open-source thinking / reasoning models													
Qwen3-VL-8B-Thinking (Bai et al., 2025a)	32	32.1	35.6	14.0	100.0	35.1	50.8	10.5	34.9	24.7	40.6	30.3	43.8
VideoChat-R1.5-7B (Yan et al., 2025)	32	29.4	45.2	7.0	100.0	47.0	47.1	9.5	29.3	17.0	40.6	23.6	33.1
Video-R1-7B (Feng et al., 2025)	32	25.1	34.2	0.0	100.0	32.8	46.5	14.3	25.6	15.1	35.3	24.7	32.5
Open-source standard models													
Qwen3.5-35B-A3B	2 fps	38.4	51.1	26.1	91.3	57.5	51.9	16.2	40.4	33.0	36.3	27.5	40.2
InternVL3.5-8B	32	31.3	42.0	9.0	100.0	52.2	55.1	13.3	30.7	26.0	32.4	26.4	27.2
MiniCPM-o-4.5	1 fps	29.2	36.5	15.7	82.6	34.3	50.3	8.6	31.7	15.1	34.2	31.5	33.5
Qwen2.5-VL-72B (Bai et al., 2025b)	32	34.0	47.5	3.0	95.7	49.3	50.3	15.2	38.2	30.1	40.6	27.0	42.6
InternVL3-8B (Zhu et al., 2025)	32	30.1	42.5	18.7	91.3	42.5	43.3	5.7	32.0	20.2	31.7	26.4	27.8
LLaVA-Video-72B (Zhang et al., 2024b)	32	28.9	49.3	15.0	26.1	47.0	45.5	14.3	25.9	21.5	31.7	24.2	24.5
VideoLLaMA3-7B (Zhang et al., 2025)	32	28.4	36.1	33.7	78.3	30.6	46.0	16.2	26.4	17.0	22.3	27.0	24.7
LLaVA-Video-7B	8	27.9	32.0	26.3	100.0	31.3	49.2	9.5	28.6	21.5	18.0	27.0	21.7
Video-LLaVA-7B (Lin et al., 2024)	8	27.6	51.1	4.7	8.7	59.7	47.1	3.8	25.6	23.7	28.4	21.3	25.6
MiniCPM-V-2.6 (Yao et al., 2024)	32	27.1	35.6	18.7	100.0	44.0	52.4	9.5	23.5	17.0	25.2	24.7	22.3
Qwen2.5-VL-7B	32	27.0	40.6	0.7	95.7	38.8	43.3	14.3	25.9	15.1	43.9	25.8	35.3
LLaVA-OV-7B-1Frame (Li et al., 2024a)	1	25.4	25.1	28.3	87.0	36.6	42.8	9.5	25.7	17.9	15.8	23.0	27.8
Qwen3-VL-8B	32	25.2	35.6	12.3	95.7	41.8	44.9	11.4	20.1	17.6	31.3	20.2	25.4
Video-ChatGPT-7B (Maaz et al., 2024)	100	21.0	33.3	30.7	0.0	47.8	1.1	7.6	18.1	19.6	20.5	14.0	12.4

Event counting, ordering, and premise verification remain challenging. GPT-5.5 reaches 68.9% on event existence and 78.4% on reappearance/disappearance, but only 18.3% on event counting and 24.8% on exact ordering. For Count, always predicting the most frequent answer, "2", yields 45.3%, compared with 25.0% for uniform guessing. The examples in Appendix G.4 illustrate the errors: most models shown count two rather than four apple reappearances, or report a woman's state at the supposed return of a pouch that never reappears. HDA-S and HDA-E distinguish two forms of premise verification: GPT-5.5 scores 31.7% and 50.9%, whereas Kimi-K2.6 scores 60.6% and 47.4%, respectively (Appendix G.2).

Frame-budget sensitivity. For four models, we compare 16, 32, and 64 uniformly sampled frames while holding prompts, decoding, answer extraction, and benchmark items fixed. The range of overall accuracy across these settings is 0.3–2.5 percentage points per model. Increasing the frame budget yields no consistent improvement (Appendix F.1), and substantial difficulties persist across the tested budgets.

Closed-source models lead in overall accuracy. Closed-source models average 39.8% accuracy. Qwen3.5-35B-A3B leads the evaluated open-source models at 38.4%, compared with 34.0% for Qwen2.5-VL-72B. InternVL3.5-8B and MiniCPM-o-4.5 score 31.3% and 29.2%, respectively. All remain substantially below the human baseline. The effect of using Qwen3-VL for filtering is discussed in Appendix D.1.

Reasoning-style variants show model-dependent gains. Qwen3-VL-8B-Thinking outperforms Qwen3-VL-8B in both overall accuracy and HDA. VideoChat-R1.5-7B and Video-R1-7B remain below several standard models, indicating that reasoning-style variants do not uniformly improve performance under the evaluated settings.

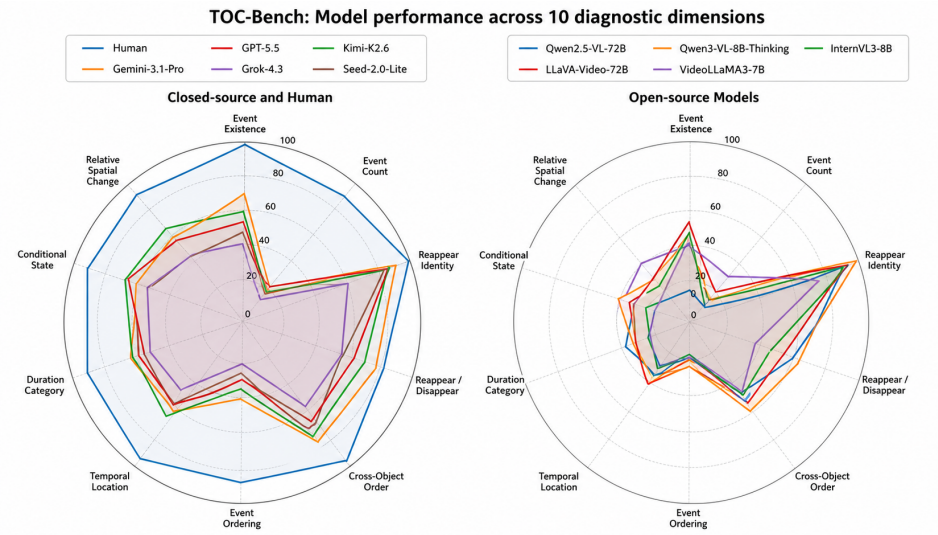

Figure 6: Radar visualization of model performance across the 10 diagnostic dimensions of TOC-Bench. The left plot compares human performance with representative closed-source models, while the right plot compares representative open-source models from the base 23-model evaluation. The visualization highlights the human–model gap and shows that event counting and event ordering remain difficult across model families.

Implications for model design. The results motivate better representations of object-event histories and more reliable verification of temporal premises. Object-centric memory, tracking-aware visual tokens, and supervision of visibility and events are promising directions. Question-conditioned adaptive sampling may also improve access to relevant evidence. TOC-Bench provides complementary dimensions for evaluating these approaches in short- to medium-length videos under explicitly reported input settings.

These dimension-level profiles can guide targeted model development. Repeated-event counting and ordering can assess an object-memory module, while HDA-S/E and valid-premise accuracy jointly assess a premise-verification strategy. Reporting these scores alongside Overall helps distinguish broad improvement from gains concentrated in the largest dimensions. For question-conditioned sampling, the uniform-sampling results provide a reference for testing whether selecting more relevant frames helps beyond increasing their number. Such comparisons can reveal which aspects of reasoning over object histories improve and which errors remain.

Additional evaluation breakdowns, including tier-level accuracy, format-level accuracy, hallucination-bucket results, and model-by-dimension visualizations, are provided in Appendix G.

5 Conclusion

We introduce TOC-Bench, a diagnostic benchmark for evaluating temporal object consistency in Video-LLMs. Grounding questions in object tracks and event timelines, it tests whether models can track object identity, state, and persistence through occlusion, reappearance, and interactions with other objects. Temporal-necessity filtering reduces shortcut-solvable questions, while hallucination-aware variants probe unsupported premises about objects or events. Experiments reveal gaps in repeated-event counting, exact ordering, and premise verification, motivating stronger object-centric memory and temporal reasoning. The verified set prioritizes diagnostic quality over scale; future work can expand its coverage and explore adaptive selection of temporal evidence. Potential prior exposure to public source videos, dependence on a single filtering model, and residual annotation uncertainty remain limitations.

AI USE STATEMENT

Qwen3-VL-8B-Instruct is used for object proposals and temporal shortcut probes, and GPT-5.4-mini for surface realization of structured QA records. Their roles and constraints are described in Section 3.1 and Appendices A–D. An AI writing assistant also assisted with language editing, literature lookup, interpretation of recorded results, and preparation of LaTeX tables. Reported model results come from the recorded evaluations.

ETHICS STATEMENT

TOC-Bench uses existing public videos and new QA annotations for academic evaluation. The videos retain their original source licenses: Perception Test data use CC BY 4.0, MOSE and OVIS use CC BY-NC-SA 4.0, and Charades retains its source-specific research/evaluation conditions. Annotation or code licensing does not relicense these videos. ReID denotes within-video object identity, not real-world person identification; no demographic or biometric labels are added. Verification and human evaluation were conducted internally, without external crowdsourcing or collection of participant personal data. The benchmark inherits source coverage and privacy limitations and is intended for model diagnosis rather than deployment or individual identification.

REPRODUCIBILITY STATEMENT

Section 3.1 and Appendices A–D describe construction, filtering, and expert verification. Section 4 and Appendix F specify evaluation inputs, answer spaces, extraction, and scoring. Appendix G provides detailed results and statistical analyses. The anonymous entry point for code and benchmark resources is `https://anonymous.4open.science/r/toc_bench-1355`.

REFERENCES

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

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo 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, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

ByteDance Seed Team. Seed2.0. `https://seed.bytedance.com/en/seed2`, 2026. Official model page for the Seed2.0 series, including Pro, Lite, and Mini. Accessed: 2026-05-02.

Mu Cai, Reuben Tan, Jianrui Zhang, Bocheng Zou, Kai Zhang, Feng Yao, Fangrui Zhu, Jing Gu, Yiwu Zhong, Yuzhang Shang, Yao Dou, Jaden Park, Jianfeng Gao, Yong Jae Lee, and Jianwei Yang. TemporalBench: Benchmarking fine-grained temporal understanding for multimodal video models. *arXiv preprint arXiv:2410.10818*, 2024.

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

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. InternVL: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 24185–24198, 2024.

Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, et al. VideoLLaMA 2: Advancing spatial-temporal modeling and audio understanding in Video-LLMs. *arXiv preprint arXiv:2406.07476*, 2024.

Daniel Cores, Michael Dorkenwald, Manuel Mucientes, Cees G. M. Snoek, and Yuki M. Asano. TVBench: Redesigning video-language evaluation. arXiv preprint arXiv:2410.07752, 2024.

Henghui Ding, Chang Liu, Shuting He, Xudong Jiang, Philip HS Torr, and Song Bai. MOSE: A new dataset for video object segmentation in complex scenes. In *ICCV*, 2023.

Shuangrui Ding, Rui Qian, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Yuwei Guo, Dahua Lin, and Jiaqi Wang. SAM2Long: Enhancing SAM 2 for long video segmentation with a training-free memory tree. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13614–13624, 2025.

Xinyu Fang, Kangrui Mao, Haodong Duan, Xiangyu Zhao, Yining Li, Dahua Lin, and Kai Chen. MMBench-Video: A long-form multi-shot benchmark for holistic video understanding. *Advances in Neural Information Processing Systems*, 37:89098–89124, 2024.

Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Junfei Wu, Xiaoying Zhang, Benyou Wang, and Xiangyu Yue. Video-R1: Reinforcing video reasoning in MLLMs. *arXiv preprint arXiv:2503.21776*, 2025.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-MME: The first-ever comprehensive evaluation benchmark of multi-modal LLMs in video analysis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 24108–24118, 2025.

Chaoyou Fu, Haozhi Yuan, Yuhao Dong, Yi-Fan Zhang, Yunhang Shen, Xiaoxing Hu, Xueying Li, Jinsen Su, Chengwu Long, Xiaoyao Xie, et al. Video-MME-v2: Towards the next stage in benchmarks for comprehensive video understanding. *arXiv preprint arXiv:2604.05015*, 2026.

Google. Gemini 3.1 Pro Preview. `https://ai.google.dev/gemini-api/docs/models/gemini-3.1-pro-preview`, 2026. Google AI for Developers model documentation. Accessed: 2026-05-02.

Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. TGIF-QA: Toward spatio-temporal reasoning in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2758–2766, 2017.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026, 2023.

Benno Krojer, Mojtaba Komeili, Candace Ross, Quentin Garrido, Koustuv Sinha, Nicolas Ballas, and Mahmoud Assran. A shortcut-aware video-qa benchmark for physical understanding via minimal video pairs. *arXiv preprint arXiv:2506.09987*, 2025.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. LLaVA-OneVision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024a.

Chaoyu Li, Eun Woo Im, and Pooyan Fazli. VidHalluc: Evaluating temporal hallucinations in multimodal large language models for video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13723–13733, 2025a.

Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. MVBench: A comprehensive multi-modal video understanding benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22195–22206, 2024b.

KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. VideoChat: Chat-centric video understanding. *Science China Information Sciences*, 68(10):200102, 2025b.

Shicheng Li, Lei Li, Yi Liu, Shuhuai Ren, Yuanxin Liu, Rundong Gao, Xu Sun, and Lu Hou. VITATECS: A diagnostic dataset for temporal concept understanding of video-language models. In *European Conference on Computer Vision*, pp. 331–348. Springer, 2024c.

Xinhao Li, Yi Wang, Jiashuo Yu, Xiangyu Zeng, Yuhan Zhu, Haian Huang, Jianfei Gao, Kunchang Li, Yinan He, Chenting Wang, et al. VideoChat-Flash: Hierarchical compression for long-context video modeling. *arXiv preprint arXiv:2501.00574*, 2024d.

Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-LLaVA: Learning united visual representation by alignment before projection. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pp. 5971–5984, 2024.

Pengyiang Liu, Zhongyue Shi, Hongye Hao, Qi Fu, Xueting Bi, Siwei Zhang, Xiaoyang Hu, Zitian Wang, Linjiang Huang, and Si Liu. SVCBench: A streaming video counting benchmark for spatial-temporal state maintenance, 2026. URL https://arxiv.org/abs/2603.12703.

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

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Khan. Video-ChatGPT: Towards detailed video understanding via large vision and language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12585–12602, 2024.

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

Moonshot AI. Kimi-K2.6 Model Card. https://huggingface.co/moonshotai/Kimi-K2.6, 2026. Hugging Face model card. Accessed: 2026-05-02.

OpenAI. GPT-5.5 Model Documentation. https://developers.openai.com/api/docs/models/gpt-5.5, 2026. Official API documentation. Accessed: 2026-05-02.

Viorica Patraucean, Lucas Smaira, Ankush Gupta, Adria Recasens, Larisa Markeeva, Dylan Banarse, Skanda Koppula, Mateusz Malinowski, Yi Yang, Carl Doersch, et al. Perception Test: A diagnostic benchmark for multimodal video models. *Advances in Neural Information Processing Systems*, 36: 42748–42761, 2023.

Jiyang Qi, Yan Gao, Yao Hu, Xinggang Wang, Xiaoyu Liu, Xiang Bai, Serge Belongie, Alan Yuille, Philip Torr, and Song Bai. Occluded video instance segmentation: A benchmark. *International Journal of Computer Vision*, 2022.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. URL https://arxiv.org/abs/2408.00714.

Michael S Ryoo, Honglu Zhou, Shrikant Kendre, Can Qin, Le Xue, Manli Shu, Jongwoo Park, Kanchana Ranasinghe, Silvio Savarese, Ran Xu, et al. xGen-MM-Vid (BLIP-3-Video): You only need 32 tokens to represent a video even in VLMs. *arXiv preprint arXiv:2410.16267*, 2024.

Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ivan Laptev, Ali Farhadi, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. *ArXiv e-prints*, 2016. URL http://arxiv.org/abs/1604.01753.

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

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

Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Zun Wang, Yansong Shi, et al. InternVideo2: Scaling foundation models for multimodal video understanding. In *European conference on computer vision*, pp. 396–416. Springer, 2024b.

Yi Wang, Xinhao Li, Ziang Yan, Yinan He, Jiashuo Yu, Xiangyu Zeng, Chenting Wang, Changlian Ma, Haian Huang, Jianfei Gao, et al. InternVideo2.5: Empowering video MLLMs with long and rich context modeling. *arXiv preprint arXiv:2501.12386*, 2025b.

Yuxuan Wang, Yueqian Wang, Dongyan Zhao, Cihang Xie, and Zilong Zheng. VideoHallucer: Evaluating intrinsic and extrinsic hallucinations in large video-language models. *arXiv preprint arXiv:2406.16338*, 2024c.

Bo Wu, Shoubin Yu, Zhenfang Chen, Joshua B Tenenbaum, and Chuang Gan. STAR: A benchmark for situated reasoning in real-world videos. In *Thirty-fifth Conference on Neural Information Processing Systems (NeurIPS)*, 2021.

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

xAI. Grok 4.3. `https://docs.x.ai/developers/models/grok-4.3`, 2026. Official model documentation. Accessed: 2026-05-02.

Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9777–9786, June 2021.

Xiaomi. Xiaomi MiMo-V2-Omni. `https://mimo.xiaomi.com/mimo-v2-omni`, 2026. Official model page. Accessed: 2026-05-02.

Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. Video question answering via gradually refined attention over appearance and motion. In *Proceedings of the 25th ACM international conference on Multimedia*, pp. 1645–1653, 2017.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. MSR-VTT: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5288–5296, 2016.

Shuhang Xun, Sicheng Tao, Jungang Li, Yibo Shi, Zhixin Lin, Zhanhui Zhu, Yibo Yan, Hanqian Li, Linghao Zhang, Shikang Wang, et al. RTV-Bench: Benchmarking MLLM continuous perception, understanding and reasoning through real-time video. *arXiv preprint arXiv:2505.02064*, 2025.

Ziang Yan, Xinhao Li, Yinan He, Zhengrong Yue, Xiangyu Zeng, Yali Wang, Yu Qiao, Limin Wang, and Yi Wang. VideoChat-R1.5: Visual test-time scaling to reinforce multimodal reasoning by iterative perception. *arXiv preprint arXiv:2509.21100*, 2025.

Jinyu Yang, Mingqi Gao, Zhe Li, Shang Gao, Fangjing Wang, and Feng Zheng. Track anything: Segment anything meets videos. *arXiv preprint arXiv:2304.11968*, 2023.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. MiniCPM-V: A GPT-4V level MLLM on your phone. *arXiv preprint arXiv:2408.01800*, 2024.

Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. ActivityNet-QA: A dataset for understanding complex web videos via question answering. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 9127–9134, 2019.

Liping Yuan, Jiawei Wang, Haomiao Sun, Yuchen Zhang, and Yuan Lin. Tarsier2: Advancing large vision-language models from detailed video description to comprehensive video understanding. *arXiv preprint arXiv:2501.07888*, 2025a.

Yuqian Yuan, Hang Zhang, Wentong Li, Zesen Cheng, Boqiang Zhang, Long Li, Xin Li, Deli Zhao, Wenqiao Zhang, Yueting Zhuang, et al. VideoRefer Suite: Advancing spatial-temporal object understanding with Video LLM. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 18970–18980, 2025b.

Z.AI. GLM-5V-Turbo. https://docs.z.ai/guides/vlm/glm-5v-turbo, 2026. Official developer documentation. Accessed: 2026-05-02.

Boqiang Zhang, Kehan Li, Zesen Cheng, Zhiqiang Hu, Yuqian Yuan, Guanzheng Chen, Sicong Leng, Yuming Jiang, Hang Zhang, Xin Li, et al. VideoLLaMA 3: Frontier multimodal foundation models for image and video understanding. *arXiv preprint arXiv:2501.13106*, 2025.

Yuanhan Zhang, Bo Li, Haotian Liu, Yong Jae Lee, Liangke Gui, Di Fu, Jiashi Feng, Ziwei Liu, and Chunyuan Li. LLaVA-NeXT: A strong zero-shot video understanding model, April 2024a. URL https://llava-vl.github.io/blog/2024-04-30-llava-next-video/.

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

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. InternVL3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

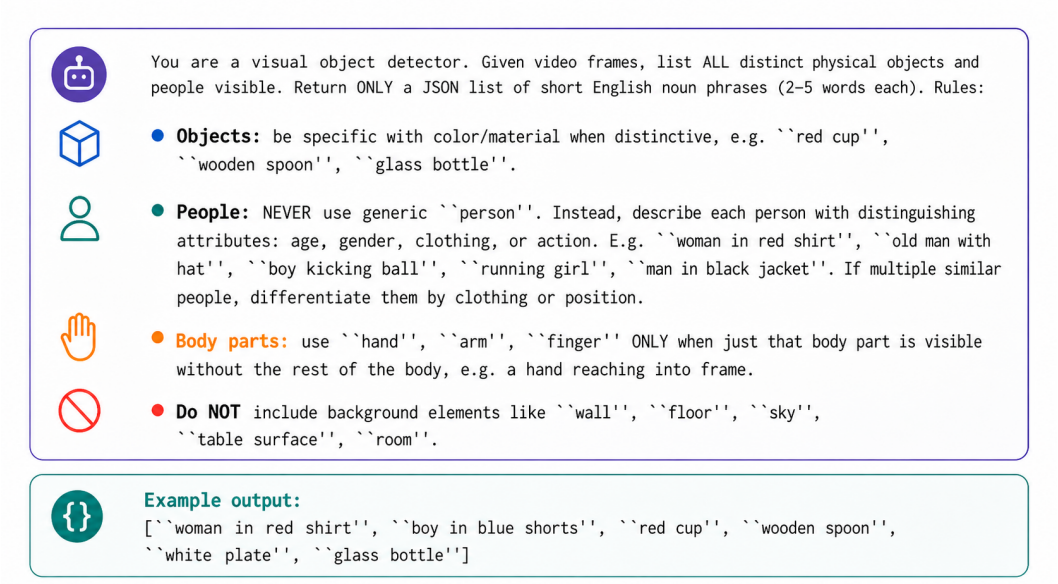

Figure 7: The system prompt used for noun-phrase extraction.

A More Details about Event Detection

A.1 Noun-Phrase Extraction

For each source video, we first sample frames at 1 fps and limit the input to a maximum of 10 frames. The sampled frames are passed to Qwen3-VL-8B-Instruct in a single multimodal turn to obtain a list of object-centric noun phrases. We use a VLM rather than a fixed object vocabulary because TOC-Bench draws from diverse video sources, where relevant objects may vary substantially across scenes.

The prompt asks the model to enumerate distinct physical objects and people, encourages discriminative attributes such as color, material, clothing, or action, and explicitly excludes background entities. The exact system prompt is shown in Fig. 7:

We keep up to 15 noun phrases per video. If parsing fails or the model returns an empty list, we use a fixed fallback prompt pool covering common categories: `person`, `hand`, `cup`, `bottle`, `ball`, `box`, `bag`, `animal`, `vehicle`, and `food`. The resulting noun phrases are cached and used as text prompts for the tracking stage.

A.2 SAM3 Tracking

We use the extracted noun phrases as text prompts for SAM3 in video-predictor mode. Each noun phrase is independently fed into SAM3, which produces per-frame masks, bounding boxes, confidence scores, and visibility flags. Tracking is performed at 3 fps. We retain masks that satisfy the confidence and area thresholds, and limit the number of tracks per video to avoid excessive noisy instances.

For each video, the raw tracking output is stored as:

$$\mathcal{O} = \{(o_i, \mathbf{m}_i^{1:T}, \text{label}_i)\}_{i=1}^{K}, \tag{2}$$

where $\mathcal{O}$ is the set of raw object tracks, K is the number of tracked objects, T is the number of tracked frames, o_i is the object ID, $\mathbf{m}_i^{1:T}$ is the per-frame binary mask sequence, and label_i is the noun phrase used to obtain the track. Each track also stores per-frame bounding boxes, visibility flags, confidence scores, and mask areas.

A.3 TRACK POST-PROCESSING

Raw SAM3 tracks may contain unstable short fragments, ID splits, and ID swaps, especially in videos with occlusion or visually similar objects. We therefore apply a lightweight post-processing step before event detection.

First, short fragments are removed if they are visible for too few frames. Second, possible ID splits are repaired by merging same-label tracks whose visible time windows do not overlap and whose positions are spatially continuous. Third, possible ID swaps are flagged when a track exhibits an abrupt position jump between nearby visible frames. We do not automatically split such tracks; instead, the detected swaps are recorded and used to reduce the track-level confidence score.

Each retained track is assigned an instance confidence score:

$$c = c_{\text{base}} \cdot \pi_{\text{split}} \cdot \pi_{\text{swap}} \cdot \beta_{\text{cov}}, \tag{3}$$

where c_{base} is the mean SAM3 confidence over visible frames, π_{split} penalizes merge operations, π_{swap} penalizes unresolved swap detections, and β_{cov} reflects temporal coverage. Tier-2 and Tier-3 reasoning units use this score to filter unreliable subject tracks, reducing the risk of generating identity-sensitive questions from unstable trajectories.

A.4 TEMPORAL EVENT DETECTION

After track post-processing, we extract temporal events from each object's mask and bounding-box sequence. The detector uses per-frame visibility, mask-area changes, object positions, and pairwise bounding-box overlap to identify object-level and pairwise events.

We detect eight per-object event types: `appear`, `enter_frame`, `exit_frame`, `partial_occlusion`, `full_occlusion`, `reappear`, `disappear`, and `state_change`. We also detect one pairwise event type, `interaction`, when two object tracks overlap for a sustained period. For occlusion events, we further store possible occluder candidates by checking nearby objects whose bounding boxes overlap with the target object around the occlusion onset.

Each event is stored with metadata needed for QA construction, including its object ID, event type, frame or frame range, timestamp, temporal-position bucket, duration bucket when applicable, and possible related objects. We also build object timelines by sorting each object's events chronologically. These timelines are later used to construct event-ordering, reappearance, event-count, and cross-object reasoning units.

A.5 VIDEO-LEVEL RETENTION

Finally, we filter videos based on whether they contain enough object-level temporal structure for QA construction. A video is retained only if it has sufficient duration, at least two tracked objects, enough meaningful temporal events, and either occlusion-related or interaction-related phenomena. Each retained video is also assigned a quality score that favors richer object-temporal dynamics, such as occlusion, reappearance, interaction, and multi-object involvement. This score is later used when sampling videos for reasoning-unit construction. Main hyperparameters used in Stage 1 are shown in Table 3.

B MORE DETAILS ABOUT QA CONSTRUCTION

This appendix provides implementation details for Stage 2, which converts the object-level temporal annotations from Stage 1 into candidate question-answer pairs. The stage consists of four steps: video sampling, reasoning-unit construction, skeleton synthesis with hallucination injection, and surface realization. A summary of the main implementation settings is provided in Table 5.

B.1 VIDEO SAMPLING

We select 6,000 videos from the Stage-1 retained pool before constructing reasoning units. Since the retained videos are not uniformly distributed over object-level temporal phenomena, we use a phenomenon-first sampling strategy. The goal is to ensure that the selected videos cover both common

Table 3: Main hyperparameters used in Stage 1 object-temporal annotation.

Component	Parameter	Value
Noun-phrase extraction	Frame sampling rate	1 fps
Noun-phrase extraction	Maximum frames per video	10
Noun-phrase extraction	Maximum noun phrases per video	15
Noun-phrase extraction	Fallback prompt pool size	10
SAM3 tracking	Tracking frame rate	3 fps
SAM3 tracking	Confidence threshold	0.3
SAM3 tracking	Minimum mask area	100 px
SAM3 tracking	Maximum tracks per video	20
Track post-processing	Minimum fragment length	5 frames or 2% of video
Track post-processing	Maximum split-repair time gap	2 s
Track post-processing	Maximum split-repair spatial jump	15% of image diagonal
Track post-processing	Swap-detection spatial jump	35% of image diagonal
Track post-processing	Swap-detection visible-frame gap	3 frames
Track post-processing	Tier-2/Tier-3 confidence cutoff	0.7
Event detection	Partial-occlusion area ratio	0.50
Event detection	Full-occlusion area ratio	0.10
Event detection	Minimum occlusion segment length	2 frames
Event detection	Reappearance confirmation	2 visible frames
Event detection	Interaction IoU threshold	0.05
Event detection	Minimum interaction length	3 frames
Video retention	Duration range	5–90 s
Video retention	Minimum tracked objects	2
Video retention	Minimum meaningful events	3
Video retention	Minimum interactions if no occlusion	2

temporal events, such as appearance and interaction, and rarer consistency-critical phenomena, such as reappearance, repeated events, cross-event chains, and high-confidence multi-object cases.

For each candidate video v, we compute a sampling score:

$$\begin{aligned} S(v) = & \underbrace{\sum_{s \in \mathcal{S}} w_s \cdot \mathbf{1}[v \text{ covers } s] \cdot \Delta_s}_{\text{phenomenon coverage}} \\ & + \underbrace{\lambda_{\text{dur}} \Delta_{\text{dur}}(b_{\text{dur}}(v))}_{\text{duration balance}} + \underbrace{\lambda_{\text{den}} \Delta_{\text{den}}(b_{\text{den}}(v))}_{\text{object-density balance}} \\ & - \underbrace{\lambda_{\text{src}} \mathbf{1}[\text{source}(v) \text{ exceeds cap}]}_{\text{source penalty}}. \end{aligned} \tag{4}$$

Here, $\mathcal{S}$ denotes the set of phenomenon slots, w_s is the slot weight, and $\mathbf{1}[v \text{ covers } s]$ indicates whether video v contains phenomenon slot s. The term $\Delta_s = \max(0, \tau_s - r_s)$ denotes the remaining deficit of slot s, where τ_s is the target ratio and r_s is the current selected ratio. Similarly, $b_{\text{dur}}(v)$ and $b_{\text{den}}(v)$ denote the duration bucket and object-density bucket of video v, and Δ_{dur} and Δ_{den} measure the remaining deficits of those buckets. The source penalty discourages selecting videos from a source whose current share has exceeded its cap, with $\lambda_{\text{src}} > 0$. Once a phenomenon or bucket reaches its target ratio, its deficit becomes zero and no longer increases the score. If all major phenomenon deficits are filled before reaching the target number of videos, the remaining videos are selected by their Stage-1 quality scores.

B.2 Reasoning Unit Construction

For each selected video, we convert its object timelines and temporal events into structured reasoning units. A reasoning unit is an intermediate representation of a possible QA item:

$$u = (\text{dim}, \text{video_id}, \text{subject}, \text{payload}), \tag{5}$$

where dim specifies the diagnostic dimension, $\mathrm{subject}$ identifies the queried object track, and $\mathrm{payload}$ stores dimension-specific information such as the anchor event, correct temporal bucket, duration category, event count, sibling object, partner object, or correct ordering relation.

For example, the apple case in Fig. 19 corresponds to video `perception_test_video_6134`, object 3. Its recorded reappearance events occur at approximately 4.667, 10.333, 20.333, and 27.667 seconds on the 3-fps tracking timeline. The event-count unit stores `subject=apple`, `event_type=reappear`, `raw_count=4`, and answer 4. The natural-language QA (ID `09f5094808ca`) and final verified item preserve this answer. This example makes the link from instance-specific events to a scored question explicit.

Reasoning units are generated by dimension-specific builders. Each builder enforces hard preconditions before emitting a unit. For example, temporal-location units require an unambiguous event occurrence; duration-category units require an event with a well-defined temporal extent; event-count units require repeated events of the same type; and cross-object order units require two temporally separated events from different object tracks. For Tier-2 and Tier-3 dimensions, the involved object tracks must pass the instance-confidence threshold introduced in Stage 1. This reduces the risk of constructing identity-sensitive questions from unstable tracks.

Across the selected 6,000 videos, this step produces 547,610 candidate reasoning units across 10 diagnostic dimensions. These units are not yet natural-language questions. Instead, they provide the structured logical basis for skeleton construction.

B.3 Question Skeleton Synthesis

The reasoning units are then rendered into question skeletons using dimension-specific templates. A skeleton is a fully specified QA record containing a question stem, answer format, correct answer, and format-specific options or statements. The correct answer is determined entirely from the structured reasoning unit, not by a language model.

TOC-Bench uses four task types: multiple-choice, statement pair, numerical, and ordering. Multiple-choice skeletons are used for bucket-style dimensions such as temporal location, duration category, relative spatial change, and conditional state. Statement-pair skeletons are used for binary judgments such as event existence, reappearance/disappearance mechanism, reappearance identity, and cross-object order. Numerical skeletons are used for event-count questions, while ordering skeletons ask the model to arrange three or four events chronologically.

Distractors are generated in a dimension-aware way. Bucket-style dimensions sample distractors from the corresponding bucket vocabulary. Statement-pair dimensions construct the contrastive statement by reversing the mechanism, identity, polarity, or event order. Numerical questions use the controlled answer set $\{2, 3, 4, \text{“5 or more”}\}$. Ordering questions present events in shuffled order while storing the correct chronological sequence.

B.4 Hallucination-aware Variant Construction

To test whether models accept unsupported premises about objects or events, we introduce hallucination-aware variants during skeleton construction. These variants are applied to selected multiple-choice dimensions: temporal location, duration category, relative spatial change, and conditional state.

We use three treatments. In variant A, the subject in the question is replaced with a plausible but nonexistent object, and the correct answer becomes a statement of the form “there is no X in this video”. In variant B, the subject exists, but the described event does not occur, and the correct answer becomes a statement of the form “X never Y”. In the distractor-only treatment, the original question remains valid, but one wrong option is replaced with a hallucination-style distractor. The relative-spatial-change dimension excludes variant B because “never moves” would conflict with the valid answer “stays in roughly the same place”.

Nonexistent-subject candidates are generated using two strategies. Most retain the original base noun but replace its modifier with a different color, material, or descriptive attribute. A smaller proportion uses phrases sampled from the global label pool whose base nouns do not appear among the current

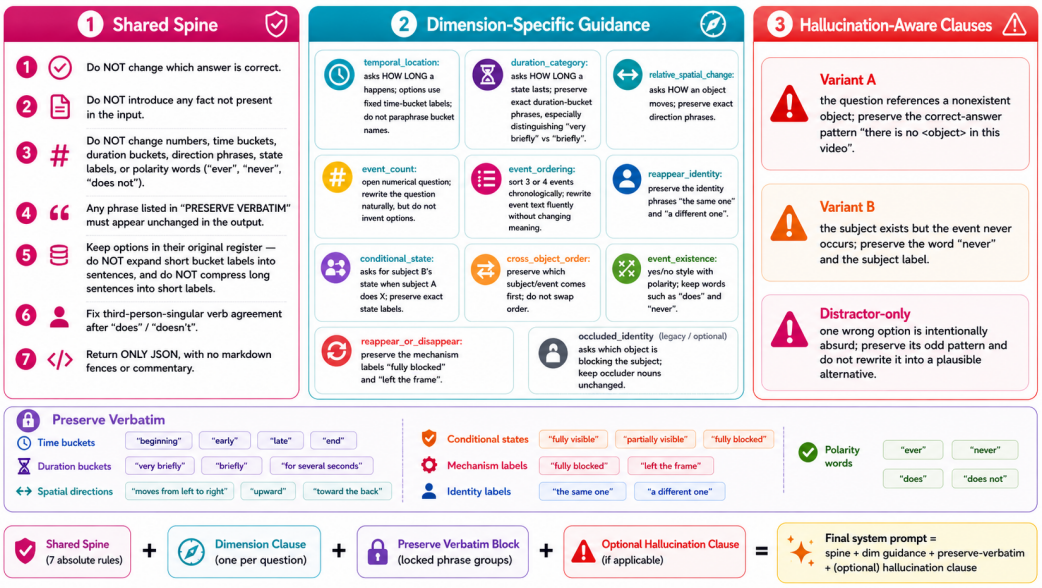

Figure 8: Dimension-aware surface-realization prompt design used in Stage 2. The system prompt consists of shared instructions, dimension-specific guidance, a preserve-verbatim block for fixed labels, and an optional hallucination-aware clause. This design allows GPT-5.4-mini to improve linguistic fluency while preserving deterministic answer semantics.

video's object labels. In both cases, candidates matching the video's object labels or the original subject label are excluded.

After hallucination injection, we apply answer balancing and a per-video cap. Answer balancing down-samples each dimension so that correct-answer labels are less biased, while keeping hallucination-tagged skeletons separate from normal answer buckets. The per-video cap prevents a small number of videos from dominating the QA pool and uses round-robin selection across dimensions to preserve diversity. After these steps, 45,527 skeletons are retained for surface realization.

B.5 Dimension-aware Surface Realization

Finally, skeletons are converted into fluent natural-language QA items using GPT-5.4-mini. The LLM is only used to polish the surface form. It does not determine the correct answer, modify the answer label, or change any structured metadata.

To reduce semantic drift, each call uses a dimension-aware system prompt. As illustrated in Fig. 8, the prompt has four components: shared instructions, dimension-specific guidance, a preserve-verbatim block, and an optional hallucination-specific clause. The shared instructions prohibit changes to the correct answer or additions of unsupported facts, require numbers and controlled labels to remain unchanged, allow minor grammatical corrections, and specify JSON output.

The dimension-specific clause explains what each dimension is asking and which parts must remain unchanged. For example, temporal-location items instruct the model that the options are fixed time-bucket labels; duration-category items require preserving the distinction between "very briefly" and "briefly"; relative-spatial-change items require preserving direction phrases; conditional-state items require preserving visibility-state labels; and ordering items instruct the model to rewrite event descriptions without changing their chronological semantics.

For dimensions with controlled answer vocabularies, the prompt also includes a `PRESERVE VERBATIM` block. This block lists phrases that must survive rewriting exactly. The locked phrases include temporal buckets, duration buckets, spatial-direction labels, conditional-state labels, reappearance identity labels, mechanism labels, and key polarity words. The main locked vocabularies are summarized in Table 4.

Table 4: Representative locked phrases used during dimension-aware surface realization. These phrases must be preserved verbatim in the polished QA item.

Dimension	Locked phrases
`temporal_location`	`beginning`, `early`, `late`, `end`
`duration_category`	`very briefly`, `briefly`, `for several seconds`, `for an extended period`
`relative_spatial_change`	`moves from left to right`, `moves from right to left`, `moves from top to bottom`, `moves from bottom to top`, `stays in roughly the same place`
`conditional_state`	`fully visible`, `partly hidden`, `fully hidden`, `not yet appeared or already gone`
`reappear_or_disappear`	`fully blocked`, `left the frame`
`reappear_identity`	`the same one`, `a different one`

Table 5: Summary of the main implementation settings in Stage 2.

Component	Setting	Value
Video sampling	Selected videos	6,000
Video sampling	Duration buckets	short / medium / long
Video sampling	Object-density buckets	sparse / moderate / dense
Video sampling	Maximum source ratio	0.45
Reasoning units	Total reasoning units	547,610
Reasoning units	Diagnostic dimensions	10
Reasoning units	Tier-2/Tier-3 confidence gate	$c \geq 0.7$
Skeletons	Retained skeletons	45,527
Skeletons	Task types	multiple-choice / statement pair / numerical / ordering
Hallucination	Variant A	nonexistent subject
Hallucination	Variant B	nonexistent event
Hallucination	Distractor-only	hallucination-style wrong option
Surface realization	Realization model	GPT-5.4-mini
Surface realization	Prompt design	shared spine + dim clause + preserve-verbatim block
Surface realization	Hallucination protection	preserve "there is no" and "never" patterns
Surface realization	Error handling	field-level rollback

For hallucination-related skeletons, an additional clause is appended to the system prompt. For variant-A hallucination questions, where the question references a nonexistent object, the prompt explicitly requires preserving the pattern "there is no ⟨object⟩ in this video" in the correct option. For variant-B hallucination questions, where the subject exists but the queried event never happens, the prompt requires preserving the word "never" and the same subject label. For normal questions that contain a hallucination-style distractor, the prompt instructs the model not to rewrite the intentionally absurd option into a plausible alternative.

After surface realization, each item is automatically checked for empty fields, duplicate options, bucket-label violations, hallucination-marker violations, numerical-answer violations, and structural errors in ordering questions. If a rewritten field violates a locked-phrase constraint, it reverts to the original skeleton text. For hallucination-marker violations, all four options revert to preserve the intended answer structure. These checks constrain wording changes while preserving the structured semantics and answer labels.

C More Details about Filtering and Verification

C.1 Three-Layer Temporal-Necessity Filter

Given the natural-language QA pool generated in Stage 2, we apply three independent filtering layers. Each layer tests whether a question can be solved without the intended form of temporally ordered object-level reasoning. A QA item is retained only if it passes all applicable layers.

The three layers are:

Layer 1: text-only filter. The model receives only the question and answer choices, without visual input. Items meeting the correct-response criterion for this setting are removed as potential shortcuts based on language priors, answer-option bias, or textual leakage.

Layer 2: single-frame filter. The model receives a single video frame together with the question. Items answered correctly across enough individual-frame probes are removed as potential static-visual shortcuts. Rather than using one random frame, we probe a set of uniformly sampled frames progressively. The probing order is temporally dispersed, so early probes cover different parts of the video. This reduces dependence on the choice of a single frame.

Layer 3: frame-shuffled filter. The model is evaluated on the original ordered frame sequence and a shuffled version of the same frames. Items answered correctly under both conditions are removed as potential frame-order shortcuts. Items solved only with the ordered input are retained. Items answered incorrectly under both conditions are also retained because this probe does not establish shortcut success for them.

C.2 Format-Aware Answer Parsing

Because TOC-Bench contains multiple task formats, the filtering model output is parsed in a format-aware manner. For multiple-choice questions, the prediction must be one of `A`, `B`, `C`, or `D`. For statement-pair questions, the prediction must be `A` or `B`. For ordering questions, the prediction is parsed as an ordered sequence of event labels, such as `C,A,B`. For numerical questions, the prediction is normalized into one of the controlled answers: `2`, `3`, `4`, or `5 or more`.

This format-aware parsing avoids treating free-form explanations as valid answers and ensures that the filtering decision is deterministic. In particular, numerical answers are strictly matched to the controlled answer set, and ordering answers must match the full chronological label sequence.

C.3 Layer Combination

After the three filter layers are run independently, their outputs are combined by intersection. An item enters the filtered pool only if it passes all applicable layers:

$$\mathrm{Pass}(q) = \mathrm{Pass}_{\mathrm{text}}(q) \wedge \mathrm{Pass}_{\mathrm{single}}(q) \wedge \mathrm{Pass}_{\mathrm{shuffled}}(q). \tag{6}$$

The combined file stores the retained items together with their layer-level results. In our construction, the three-layer filter reduces the generated QA pool from 45,527 items to 17,900 filtered QA pairs. These filtered items serve as the candidate pool for subsequent human verification.

C.4 Sampling for Human Verification

After filtering, we sample a subset from the 17,900 filtered QA pairs for human verification. The purpose of this step is to obtain a high-quality evaluation set while preserving diagnostic coverage across dimensions. Sampling is performed at the QA-item level rather than the video level, so different QA items may still come from the same source video.

We use dimension-stratified sampling to preserve the diagnostic structure of TOC-Bench. We retain as many items as possible from smaller dimensions to maintain statistical power, while downsampling larger dimensions with per-dimension caps. This prevents high-frequency dimensions from dominating the final benchmark and ensures that rare but important Tier-2 and Tier-3 dimensions remain represented.

In total, we sample 3,000 QA items for human verification. Five experts inspect disjoint 600-item partitions, comparing the corresponding videos, questions, answer candidates, and keyed answers. Minor issues, such as unclear wording, duplicated options, or slight reference ambiguity, are manually revised when the intended visual evidence and answer remain valid. Items are removed only when they contain severe evidence or label problems, such as tracking errors, insufficient visual support, incorrect event labels, wrong answer labels, or a clear mismatch between the QA item and the video. After human verification, TOC-Bench contains 2,323 high-quality QA pairs.

The rejection rate is 677/3,000 (22.6%). Expert verification checks answer validity, while the separate three-solver baseline measures task solvability. Because verification partitions are disjoint, this process does not estimate inter-annotator agreement or the residual rate of answer-affecting errors among accepted items.

The filters provide probe-based shortcut screening rather than a guarantee of temporal necessity for every retained item. Nonexistent-subject questions primarily test visual grounding; event-based questions target reasoning over object histories. Human verification checks whether each item's evidence and answer are supported by the video.

D Model-dependence and Bias Control

TOC-Bench uses foundation models as scalable tools during benchmark construction. We separate model-assisted candidate generation from final answer verification. Models propose objects and tracks, probe shortcuts, and refine wording; structured object-event records determine candidate answers, which are retained only after rule-based checks and human verification.

D.1 Filtering-model dependence

Qwen3-VL-8B-Instruct is used for temporal-necessity filtering. Items that meet the removal criteria under its text-only, single-frame, or frame-shuffled probes are excluded. The retained set may therefore be relatively harder for Qwen-family models, and their results should be interpreted with this selection dependence in mind.

All candidates are screened under the same model-specific probing protocol. Rule-based checks and human verification determine which items are retained after shortcut screening. The filtering model and the final validation stage thus serve distinct roles in benchmark construction.

D.2 Construction-model dependence

Other upstream models are used to improve candidate quality rather than to determine final answers. Qwen3-VL-8B-Instruct proposes object noun phrases, SAM3 produces candidate object tracks, and GPT-5.4-mini performs surface realization from structured question skeletons. These models do not independently decide the ground-truth labels. Each reasoning unit fixes the queried subject, diagnostic dimension, answer type, correct answer, and admissible distractors before natural-language generation. GPT-5.4-mini only improves the wording of a pre-specified QA record, while object references, answer labels, temporal buckets, numerical answers, and ordering relations remain locked by the skeleton.

Similarly, SAM3 tracks provide candidate evidence rather than automatically trusted labels. Low-quality, fragmented, or ambiguous tracks are filtered before event derivation. Human verification then removes items with tracking errors, insufficient visual evidence, or incorrect answers, while allowing minor wording or option issues to be corrected when the evidence and answer remain valid. Thus, foundation models support scalable candidate construction, with structured records, rule-based checks, and human verification governing final item retention.

E More Details about Dataset Statistics

E.1 Source-Dimension Coverage

Figure 9 shows the distribution of diagnostic dimensions across source datasets. Although Charades and Perception Test dominate the final verified set, the retained QA items span multiple dimensions across sources. MOSE and OVIS contribute fewer final items, but their occlusion-heavy videos are useful for constructing and validating rare object-level temporal phenomena during the benchmark construction process.

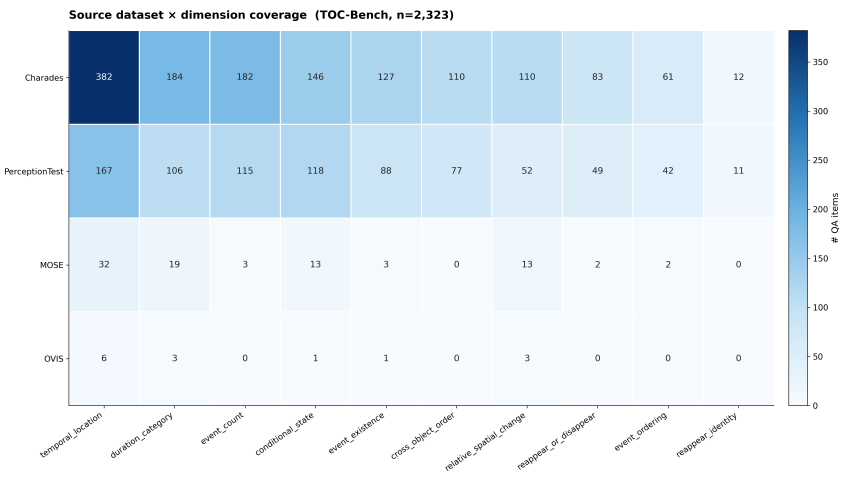

Figure 9: Source dataset by diagnostic-dimension coverage in TOC-Bench. Charades and Perception Test contribute most final QA items, while MOSE and OVIS provide additional occlusion-heavy cases. The heatmap shows that the final benchmark covers multiple diagnostic dimensions across source datasets rather than relying on a single source-dimension pair.

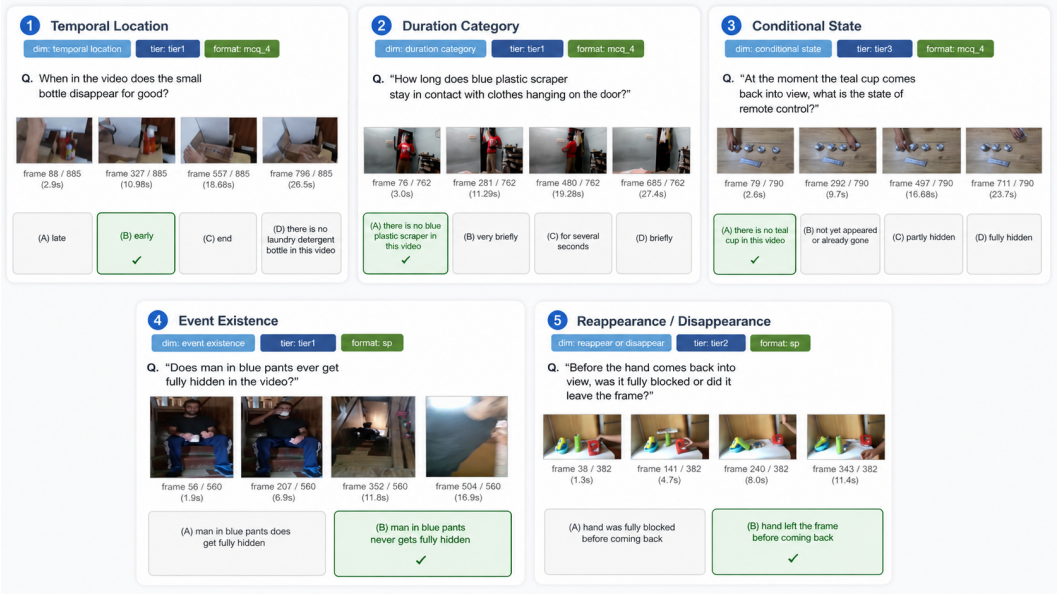

Figure 10: Dimension-wise qualitative examples from TOC-Bench, Part I. The examples cover temporal location, duration category, conditional state, event existence, and reappearance/disappearance. They illustrate how TOC-Bench tests object-level temporal reasoning over disappearance timing, duration estimation, conditional object state, event existence, and occlusion-versus-leaving-frame distinctions.

E.2 Dimension-wise Qualitative Examples

To make the diagnostic dimensions more concrete, we provide representative visual QA examples for all 10 dimensions of TOC-Bench. Each example shows the dimension, tier, task format, natural-language question, sampled video frames with timestamps, and answer candidates with the correct answer highlighted. These examples illustrate how TOC-Bench grounds questions in object-level temporal evidence rather than relying only on video-level captions or free-form object mentions.

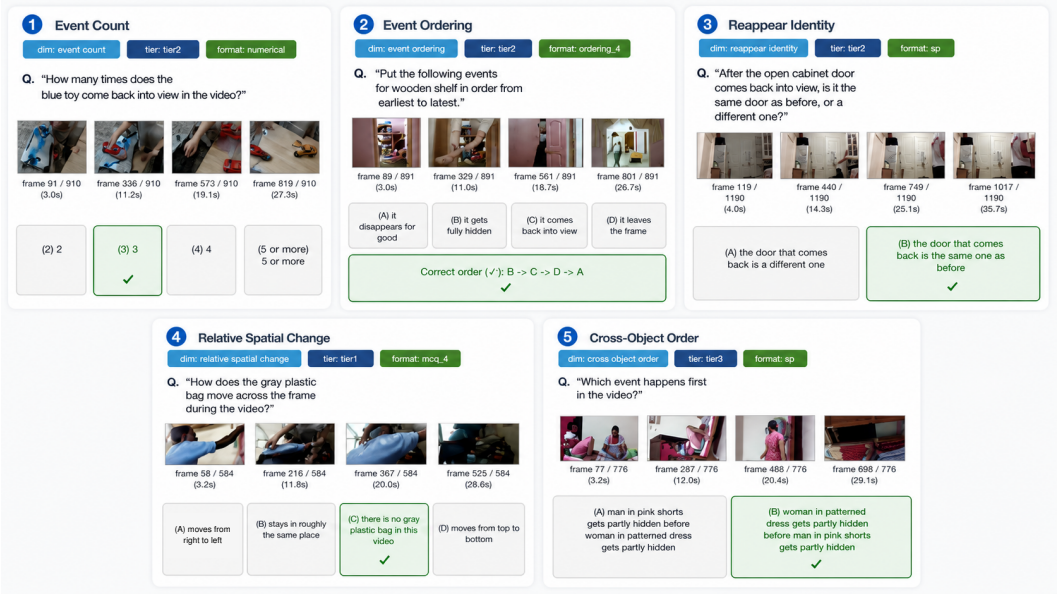

Figure 11: Dimension-wise qualitative examples from TOC-Bench, Part II. The examples cover event count, event ordering, reappearance identity, relative spatial change, and cross-object order. They illustrate repeated-event accumulation, chronological reconstruction, identity judgments, spatial change, and temporal relations between objects.

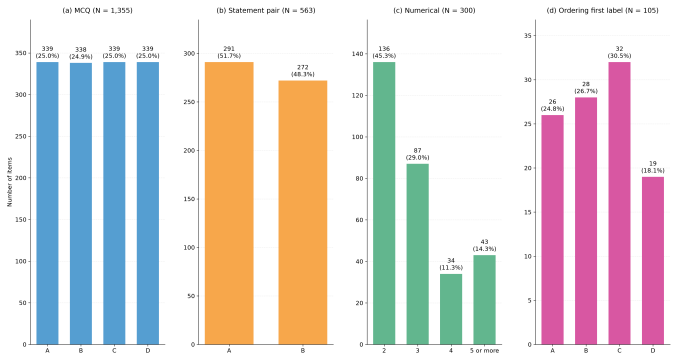

Figure 12: Answer-balance audit across question formats. Panels (a)–(c) show correct-answer distributions for multiple-choice, statement-pair, and numerical questions; panel (d) shows the first label of the correct ordering sequence.

Together, Figs. 10 and 11 show that the dimensions in TOC-Bench are not merely different surface templates. They correspond to distinct temporal object-consistency skills, including locating when an object event occurs, estimating event duration, verifying event existence, distinguishing occlusion from leaving the frame, counting repeated events, and reasoning over event order within or across object tracks.

E.3 Answer-Balance Analysis

Figure 12 reports answer-label distributions. Four-way multiple-choice labels are nearly uniform. Aggregating the four statement-pair dimensions gives 291 A and 272 B labels (51.7%/48.3%). Numerical answers are unevenly distributed: 136, 87, 34, and 43 items have answers 2, 3, 4, and "5 or more", respectively. Always predicting 2 therefore yields 45.3% accuracy, whereas uniform guessing over the four valid answers yields 25.0%. We retain uniform guessing as the common

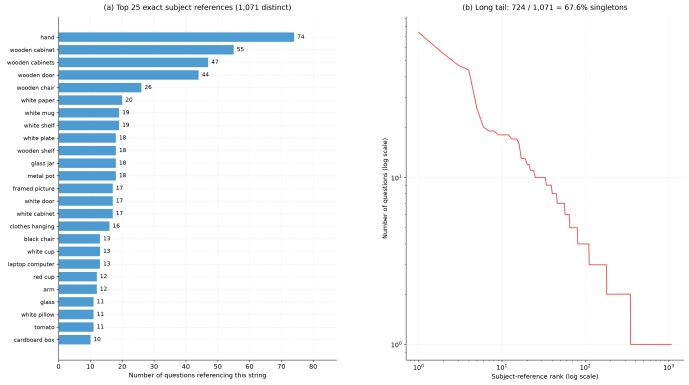

Figure 13: Subject-reference diversity in TOC-Bench. The left panel shows the most frequent subject references, while the right panel shows the distribution over 1,071 distinct subject-reference strings. Counts use exact strings, retaining singular/plural variants separately.

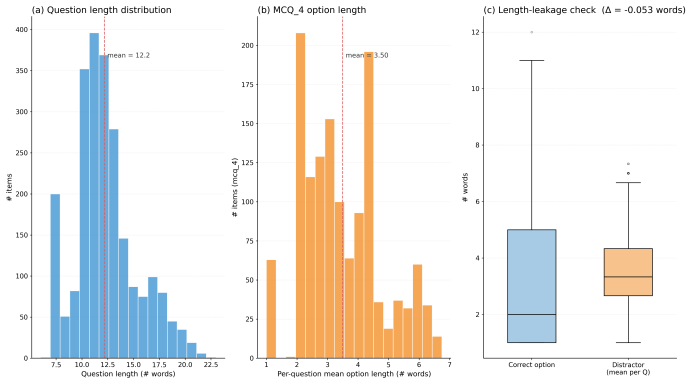

Figure 14: Length-leakage diagnostics for TOC-Bench. The analysis compares question lengths, multiple-choice option lengths, and the length difference between correct options and distractors. The mean length difference between correct options and distractors is -0.053 words.

chance reference and report the majority-answer baseline to contextualize Count performance. The ordering panel summarizes only the first label; scoring requires the complete correct permutation.

E.4 Subject-Reference Diversity

Figure 13 shows a long tail of subject references. The benchmark contains 1,071 distinct reference strings, of which 724 (67.6%) occur once. Frequent references include hands, cabinets, doors, and chairs, alongside many less frequent object descriptions used to identify the queried instances.

E.5 Length-Leakage Diagnostics

Figure 14 summarizes question and option lengths. Questions average 12.2 words, and the per-question mean option length is 3.50 words. Correct options and distractors differ by only -0.053 words in mean length, although their distributions differ. The plot characterizes this surface property of the QA items rather than testing a length-based answering strategy.

Exact dimension sizes and ReID balance. In the order Exist, Count, ReID, R/D, X-Order, Order, T-Loc, Dur, Cond, and Move, the final dimensions contain 219, 300, 23, 134, 187, 105, 587, 312, 278, and 178 items. Overall weights dimension accuracies by these sample sizes, while per-dimension scores describe performance on each diagnostic task. ReID has semantic same/different counts of

13/10 and answer-position A/B counts of 11/12. Its majority-semantic-class and majority-position baselines are therefore 56.5% and 52.2%, respectively. One item changes ReID accuracy by 4.35 percentage points, and the subset contributes 0.99% of the weight in Overall. It is therefore useful for small-scale diagnosis but not precise model ranking.

The other SP dimensions have A/B counts of Exist 112/107, R/D 80/54, and X-Order 88/99. Appendix G.5 reports position-balanced accuracy. These checks distinguish semantic balance from option-position balance and rule out constant-label prediction as an explanation of all-correct ReID results.

F Evaluation Protocol and Frame Sampling

TOC-Bench reports the input sampling and preprocessing adopted for each model and standardizes output instructions, answer extraction, and scoring. Most models use 32 uniformly sampled frames. LLaVA-Video-7B and Video-LLaVA-7B use 8, LLaVA-OV-7B-1Frame uses 1, and Video-ChatGPT-7B uses 100. Qwen3.5-35B-A3B uses 2 fps, InternVL3.5-8B uses 32 frames, and MiniCPM-o-4.5 uses 1 fps. All settings are given in Table 2.

Input sampling and preprocessing follow the corresponding model documentation. Task instructions, answer formats, extraction, and scoring follow the TOC-Bench protocol. This preserves model-specific input conventions while making the frame budget explicit; cross-model comparisons do not equate visual computation.

Models receive video evidence and questions, without construction tracks, masks, or event timelines. Table 6 summarizes the scored answer spaces. These conventions apply across models; parsing and evaluation failures remain incorrect in the reported accuracy.

Table 6: Answer spaces and scoring used in evaluation.

Format	Valid answer space	Correctness criterion
Multiple choice	`A`, `B`, `C`, `D`	Exact answer-label match
Statement pair	`A`, `B`	Exact answer-label match
Numerical count	`2`, `3`, `4`, `5 or more`	Match after normalization to the controlled answer set
Event ordering	A permutation of three or four event labels	Exact match of the complete sequence

Question-conditioned adaptive sampling is a model-side direction for improving access to relevant temporal evidence. The reported uniform-sampling settings provide reference points for evaluating such methods in future work.

F.1 Frame-Budget Sensitivity

We compare four models at 16, 32, and 64 uniformly sampled frames, holding prompts, decoding, answer extraction, and benchmark items fixed within each model. The full results appear below. Across these settings, each model's overall accuracy varies by 0.3–2.5 percentage points; three models peak at 16 frames, while Qwen2.5-VL-7B peaks at 32. Increasing the number of uniformly sampled frames does not consistently improve overall accuracy. This analysis does not identify an internal model mechanism, and individual items may still suffer from missing temporal evidence.

Table 7: Within-model frame-budget sensitivity. All values are accuracy (%). Only the uniformly sampled frame count changes within each model.

Model	Frames	Overall	Exist	Count	ReID	R/D	X-Order
VideoLLaMA3-7B	16	28.7	36.5	36.3	78.3	29.1	43.9
VideoLLaMA3-7B	32	28.4	36.1	33.7	78.3	30.6	46.0
VideoLLaMA3-7B	64	28.4	33.8	37.3	73.9	29.9	43.9
InternVL3-8B	16	32.6	41.1	25.3	100.0	50.0	43.3
InternVL3-8B	32	30.1	42.5	18.7	91.3	42.5	43.3
InternVL3-8B	64	31.6	44.3	21.7	100.0	45.5	50.3

Model	Frames	Overall	Exist	Count	ReID	R/D	X-Order
Qwen2.5-VL-7B	16	26.4	41.6	1.3	91.3	41.0	43.3
Qwen2.5-VL-7B	32	27.0	40.6	0.7	95.7	38.8	43.3
Qwen2.5-VL-7B	64	26.9	39.3	0.7	100.0	34.3	43.3
Qwen3-VL-8B	16	26.1	35.2	11.7	100.0	44.8	44.9
Qwen3-VL-8B	32	25.2	35.6	12.3	95.7	41.8	44.9
Qwen3-VL-8B	64	25.6	33.8	14.7	95.7	39.6	47.1

Table 8: Within-model frame-budget sensitivity. All values are accuracy (%). Only the uniformly sampled frame count changes within each model.

Model	Frames	Order	T-Loc	Dur	Cond	Move	HDA
VideoLLaMA3-7B	16	15.2	27.1	16.0	24.8	25.3	24.7
VideoLLaMA3-7B	32	16.2	26.4	17.0	22.3	27.0	24.7
VideoLLaMA3-7B	64	17.1	26.4	17.0	23.0	24.7	24.1
InternVL3-8B	16	9.5	33.9	23.1	34.2	25.3	28.6
InternVL3-8B	32	5.7	32.0	20.2	31.7	26.4	27.8
InternVL3-8B	64	8.6	29.5	22.1	33.5	28.1	27.2
Qwen2.5-VL-7B	16	12.4	24.5	13.1	42.8	25.3	33.9
Qwen2.5-VL-7B	32	14.3	25.9	15.1	43.9	25.8	35.3
Qwen2.5-VL-7B	64	13.3	25.4	18.3	43.5	26.4	35.5
Qwen3-VL-8B	16	11.4	21.8	18.3	34.2	19.7	27.8
Qwen3-VL-8B	32	11.4	20.1	17.6	31.3	20.2	25.4
Qwen3-VL-8B	64	14.3	19.8	17.9	31.7	21.3	26.6

G Additional Evaluation Results

G.1 Tier-Level and Format-Level Results

Table 9 provides a detailed breakdown by difficulty tier and task format. Tier 2, which includes identity-sensitive and repeated-event questions about the same object, is challenging for many models. Raw accuracy is generally higher on statement-pair questions than on numerical counting and event ordering, consistent with the main-text results. The format-specific baselines in Section 4 and Appendix E provide reference points for these comparisons.

G.2 Hallucination-Bucket Results

Table 10 compares valid- and false-premise performance. Kimi-K2.6 and Grok-4.3 outperform GPT-5.5 on nonexistent-subject questions, whereas GPT-5.5 achieves higher accuracy on clean and distractor-only items. Seed-2.0-Lite performs better on nonexistent-event questions than on nonexistent-subject questions. These profiles motivate reporting HDA-S/E alongside valid-premise accuracy. HDA measures correctness on false-premise items; false rejection on valid-premise items is a separate behavior that aggregate distractor-only accuracy does not identify.

The Pearson and Spearman correlations between HDA-S and HDA-E across these 23 models are 0.69 and 0.70, respectively. GPT-5.5 scores 31.7% and 50.9% on the two subsets, compared with 60.6% and 47.4% for Kimi-K2.6. These correlations summarize the association between model scores on the two false-premise subsets.

G.3 Additional Visualizations

The following visualizations cover the base 23-model evaluation. Results for three additional models appear in Table 2; their tier-level, format-level, and HDA-S/E breakdowns are not included here.

Model-by-dimension behavior. Figure 15 summarizes per-dimension accuracy for the base 23-model evaluation. Human accuracy is consistently high across all 10 dimensions, while each model family falls substantially below the human baseline on several tasks. Event counting and event

Table 9: Tier-level and format-level results for the base 23-model evaluation on TOC-Bench. Tier 1 focuses on single-object temporal reasoning, Tier 2 requires cross-time consistency for one object, and Tier 3 requires cross-object temporal reasoning. MCQ denotes four-way multiple-choice questions, SP denotes statement-pair questions, Num denotes numerical event-count questions, and Ord-3/Ord-4 denote ordering questions with three or four events.

Model	Tier 1	Tier 2	Tier 3	MCQ	SP	Num	Ord-3	Ord-4
Human	92.4	81.9	89.7	91.9	87.7	84.0	80.8	79.2
Closed-source models								
GPT-5.5	50.4	36.8	50.3	45.6	70.2	18.3	19.2	30.2
Kimi-K2.6	44.3	34.7	59.4	44.8	63.2	19.3	23.1	22.6
Gemini-3.1-Pro-Preview	45.5	32.7	52.9	44.7	60.4	18.7	21.2	13.2
Grok-4.3	43.7	34.5	55.5	44.5	60.0	19.7	19.2	15.1
Seed-2.0-Lite	44.1	32.7	47.3	41.8	55.6	24.3	23.1	20.8
GLM-5V-Turbo	39.3	29.2	44.5	36.7	57.5	13.7	19.2	15.1
Gemini-3.1-Flash-Lite-Preview	34.8	23.8	43.0	35.3	48.3	6.0	19.2	13.2
GPT-5.4-mini	32.4	27.8	41.5	31.0	48.5	21.3	17.3	5.7
Mimo-V2-Omni	31.9	20.5	44.3	31.5	50.8	4.7	13.5	1.9
Open-source thinking / reasoning models								
Qwen3-VL-8B-Thinking	31.9	21.9	44.7	33.1	43.2	14.0	15.4	5.7
VideoChat-R1.5-7B	28.2	20.8	43.2	28.0	48.5	7.0	19.2	0.0
Video-R1-7B	24.4	14.6	39.8	25.0	40.7	0.0	23.1	5.7
Open-source standard models								
Qwen2.5-VL-72B	36.3	20.1	44.5	35.4	50.8	3.0	19.2	11.3
InternVL3-8B	30.2	24.9	36.3	28.5	44.8	18.7	9.6	1.9
LLaVA-Video-72B	28.5	23.0	37.2	25.8	46.5	15.0	21.2	7.5
VideoLLaMA3-7B	25.8	31.5	31.8	23.5	39.8	33.7	25.0	7.5
LLaVA-Video-7B	27.2	27.4	30.5	24.6	40.3	26.3	9.6	9.4
Video-LLaVA-7B	28.9	17.8	35.9	25.2	50.1	4.7	5.8	1.9
MiniCPM-V-2.6	24.2	26.3	36.1	22.5	45.8	18.7	13.5	5.7
Qwen2.5-VL-7B	25.8	16.2	43.7	27.1	43.3	0.7	25.0	3.8
LLaVA-OV-7B-1Frame	23.4	29.2	26.7	21.5	36.2	28.3	11.5	7.5
Qwen3-VL-8B	22.1	22.6	36.8	21.8	42.6	12.3	15.4	7.5
Video-ChatGPT-7B	20.4	29.2	12.7	18.4	24.7	30.7	13.5	1.9

ordering remain challenging across model families, complementing the higher raw accuracies on event existence and other statement-pair dimensions.

Tier and format effects. Figure 16 summarizes performance by difficulty tier and task format. Tier 2 groups questions requiring cross-time consistency for the same object and remains challenging for many models. The format-level view shows higher raw accuracy on statement-pair questions than on numerical counting and ordering questions. The two breakdowns provide complementary views of where errors are concentrated.

Hallucination-aware behavior. Figure 17 compares accuracy on clean questions, distractor-only questions, nonexistent-subject variants, and nonexistent-event variants. Models exhibit different profiles across these groups: strong performance on valid-premise items does not always coincide with high accuracy on false-premise items. The breakdown shows which types of premises account for a model's correct and incorrect answers.

Overall accuracy versus HDA. Finally, Fig. 18 compares Overall with HDA. Their rankings are not identical: GPT-5.5 has the highest overall accuracy, while Grok-4.3 and Kimi-K2.6 achieve higher HDA. Reporting false-premise accuracy alongside Overall makes these differences visible without collapsing the full evaluation into a single score.

Table 10: Accuracy on hallucination-related buckets for the base 23-model evaluation. Clean denotes questions without hallucination injection. Distractor-only denotes questions with a hallucination-style wrong option. Variant A is HDA-S (nonexistent subject, N=208); Variant B is HDA-E (existing subject but nonexistent event, N=285).

Model	Clean	Distractor-only	Variant A	Variant B
Human	89.3	89.3	89.4	89.5
Closed-source models				
GPT-5.5	48.2	48.3	31.7	50.9
Kimi-K2.6	44.9	39.5	60.6	47.4
Gemini-3.1-Pro-Preview	42.2	44.1	50.0	46.0
Grok-4.3	42.5	39.0	59.6	49.5
Seed-2.0-Lite	41.9	42.7	32.7	47.4
GLM-5V-Turbo	38.8	38.7	26.9	40.4
Gemini-3.1-Flash-Lite-Preview	31.9	31.6	42.8	40.0
GPT-5.4-mini	34.9	33.5	18.3	35.8
Mimo-V2-Omni	31.5	29.5	40.9	30.9
Open-source thinking / reasoning models				
Qwen3-VL-8B-Thinking	30.5	26.4	48.6	40.4
VideoChat-R1.5-7B	30.5	25.3	30.8	34.7
Video-R1-7B	25.1	19.9	27.9	35.8
Open-source standard models				
Qwen2.5-VL-72B	31.5	32.1	40.9	43.9
InternVL3-8B	31.6	29.5	20.7	33.0
LLaVA-Video-72B	31.8	27.4	18.8	28.8
VideoLLaMA3-7B	32.9	23.9	17.8	29.8
LLaVA-Video-7B	30.8	27.7	15.9	26.0
Video-LLaVA-7B	30.2	24.9	26.0	25.3
MiniCPM-V-2.6	31.9	22.7	14.9	27.7
Qwen2.5-VL-7B	26.8	21.6	33.7	36.5
LLaVA-OV-7B-1Frame	28.3	19.2	33.2	23.9
Qwen3-VL-8B	29.3	18.5	21.2	28.4
Video-ChatGPT-7B	26.5	18.4	15.9	9.8

G.4 Qualitative Failure Cases

The qualitative cases illustrate two error types exposed by TOC-Bench: undercounting repeated events and accepting unsupported temporal premises. They complement the quantitative breakdowns without estimating error prevalence or identifying internal model mechanisms.

As shown in Figs. 19 and 20, the two cases involve distinct task requirements. Counting the apple's reappearances requires distinguishing repeated visibility transitions from multiple observations of the same visible interval. In the pouch example, the model must first verify that the triggering event occurs; otherwise, selecting the woman's state gives a plausible but incorrect answer. These examples illustrate the event-accumulation and premise-verification demands behind the errors reported in Table 2.

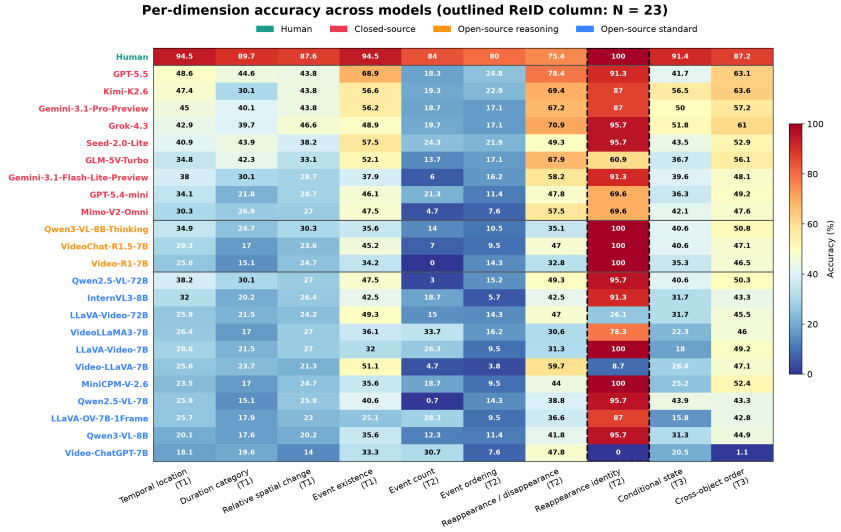

Figure 15: Model-by-dimension accuracy heatmap for the base 23-model evaluation. The outlined ReID column contains 23 items. Event counting and event ordering remain challenging across model families, while raw accuracy is generally higher on statement-pair dimensions such as event existence and cross-object order.

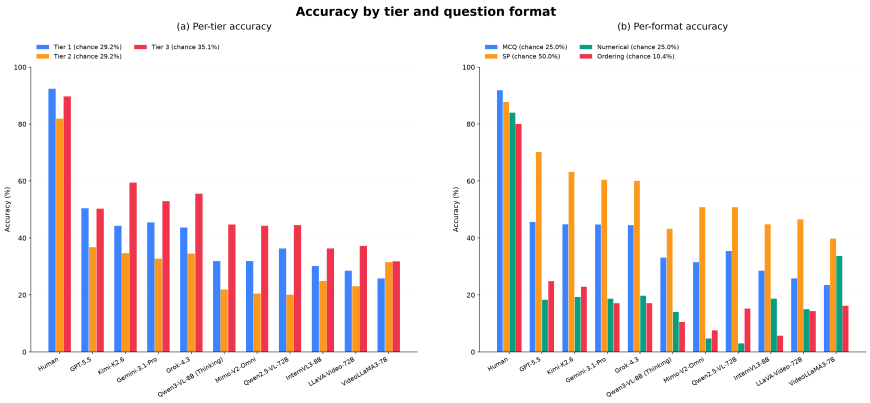

Figure 16: Tier-level and format-level accuracy breakdowns. Models generally perform better on statement-pair questions than on numerical counting and ordering formats. Legend entries report the corresponding uniform-guessing baselines; ordering accuracy pools all 105 ordering items.

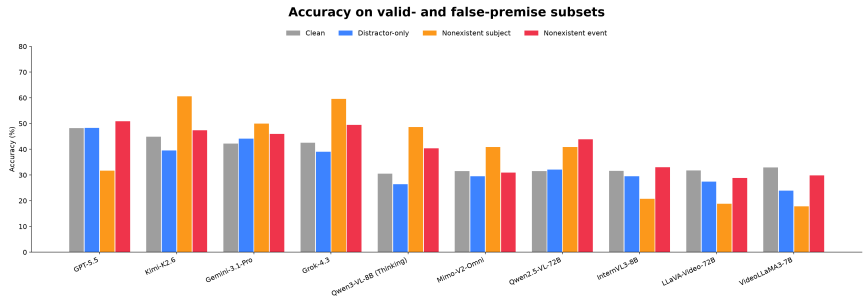

Figure 17: Hallucination-bucket accuracy across models. Models show different accuracy profiles across clean, distractor-only, nonexistent-subject, and nonexistent-event questions.

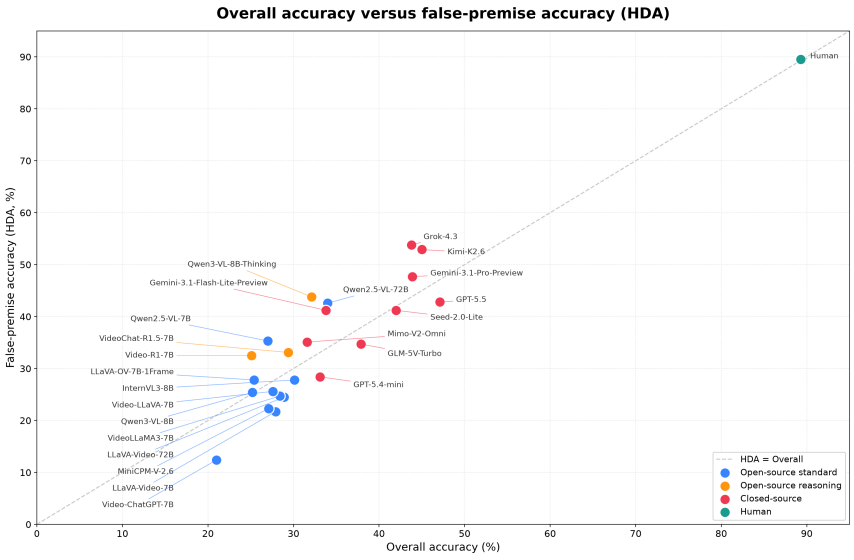

Figure 18: Overall accuracy versus hallucination diagnostic accuracy. Model rankings differ between the full benchmark and its false-premise subset.

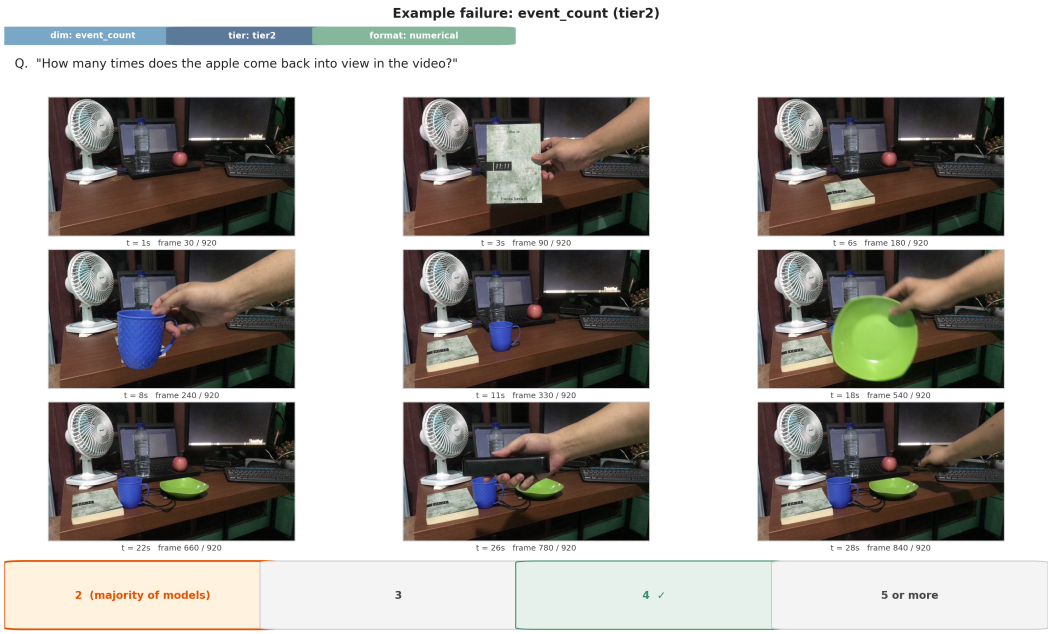

Figure 19: Failure case on **event counting**. The question asks how many times the apple comes back into view. The correct answer is 4, but most models shown predict 2. This example illustrates undercounting of repeated events associated with the same object across the video.

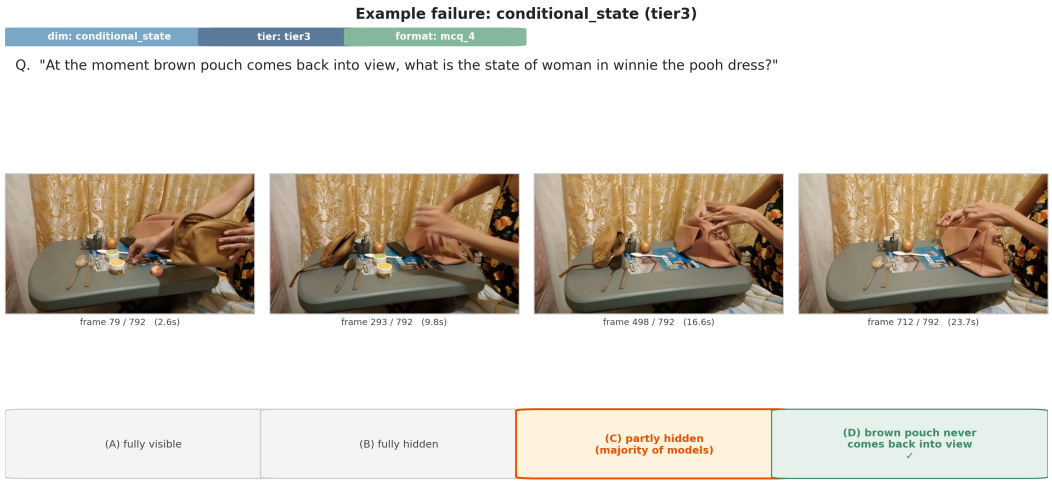

Figure 20: Failure case on **conditional state reasoning**. The question asks for the state of the woman at the moment when the brown pouch comes back into view. The correct answer is that the brown pouch never comes back into view, while the majority of models choose a plausible visual state of the woman. This example illustrates a hallucination-aware temporal failure: models tend to assume the queried event happens and answer the conditional part, instead of rejecting the false temporal premise.

G.5 Reliability and Uncertainty

Table 11 reports raw accuracy and position-balanced accuracy for all four statement-pair dimensions. Position-balanced accuracy averages recall over A and B; semantic-class balanced accuracy averages recall over same/different for ReID. A constant prediction scores 50% under the corresponding balanced measure. The near-balanced ReID counts are given in Appendix E; its small sample size limits fine-grained comparisons.

Table 11: Raw accuracy / answer-position balanced accuracy (%). Position-balanced accuracy is the mean recall over A and B; it is distinct from semantic-class balanced accuracy.

Model	ReID Acc./Pos.-Bal.	Exist Acc./Pos.-Bal.	R/D Acc./Pos.-Bal.	X-Order Acc./Pos.-Bal.
GPT-5.5	91.3/90.9	68.9/69.0	78.4/78.0	63.1/63.1
Kimi-K2.6	87.0/86.7	56.6/56.6	69.4/68.1	63.6/63.4
Gemini-3.1-Pro	87.0/86.4	56.2/56.2	67.2/65.3	57.2/57.3
Grok-4.3	95.7/95.8	48.9/49.0	70.9/69.3	61.0/60.7
Seed-2.0-Lite	95.7/95.8	57.5/57.6	49.3/48.5	52.9/52.7

We report the recorded 20,000-resample item-bootstrap analyses (seed 20260724) for six representative models. For each score cell with n QA items, each replicate samples n items with replacement and averages their binary correctness indicators. The 2.5th and 97.5th percentiles give the 95% interval. Failed or unparseable outputs remain incorrect. The tables include the sample size for every cell.

These intervals quantify QA-sample variation conditional on recorded predictions, not run-to-run/API or annotator variability. Questions can share a video, so item resampling does not capture video-level dependence. All-correct/all-wrong cells have degenerate nonparametric bootstrap intervals, such as [100,100] or [0,0]; these do not imply exact population accuracy.

Table 12: Accuracy [95% item-bootstrap interval]. These are the recorded 20,000-resample results (seed 20260724). See Appendix G.5 for limitations.

Dimension (N)	GPT-5.5	Kimi-K2.6	Qwen2.5-VL-72B
Overall(2323)	47.1 [45.1, 49.1]	45.0 [43.0, 47.1]	34.0 [32.1, 35.9]
Exist(219)	68.9 [62.6, 74.9]	56.6 [50.2, 63.0]	47.5 [40.6, 54.3]
Count(300)	18.3 [14.0, 23.0]	19.3 [15.0, 24.0]	3.0 [1.3, 5.0]
ReID(23)	91.3 [78.3, 100.0]	87.0 [73.9, 100.0]	95.7 [87.0, 100.0]
R/D(134)	78.4 [70.9, 85.1]	69.4 [61.9, 76.9]	49.3 [41.0, 57.5]
X-Ord(187)	63.1 [56.1, 70.1]	63.6 [56.7, 70.6]	50.3 [43.3, 57.8]
Ord(105)	24.8 [17.1, 33.3]	22.9 [15.2, 31.4]	15.2 [8.6, 21.9]
T-Loc(587)	48.6 [44.5, 52.5]	47.4 [43.3, 51.4]	38.2 [34.2, 42.1]
Dur(312)	44.6 [39.1, 50.0]	30.1 [25.0, 35.3]	30.1 [25.0, 35.3]
Cond(278)	41.7 [36.0, 47.5]	56.5 [50.7, 62.2]	40.6 [34.9, 46.4]
Move(178)	43.8 [36.5, 51.1]	43.8 [36.5, 51.1]	27.0 [20.8, 33.7]

Table 13: Accuracy [95% item-bootstrap interval]. These are the recorded 20,000-resample results (seed 20260724). See Appendix G.5 for limitations.

Dimension (N)	Qwen3-VL-8B-Thinking	InternVL3-8B	Video-ChatGPT 7B
Overall(2323)	32.1 [30.2, 34.0]	30.1 [28.3, 32.0]	21.0 [19.4, 22.7]
Exist(219)	35.6 [29.2, 42.0]	42.5 [36.1, 48.9]	33.3 [27.4, 39.7]
Count(300)	14.0 [10.3, 18.0]	18.7 [14.3, 23.3]	30.7 [25.7, 36.0]
ReID(23)	100.0 [100.0, 100.0]	91.3 [78.3, 100.0]	0.0 [0.0, 0.0]
R/D(134)	35.1 [26.9, 43.3]	42.5 [34.3, 50.7]	47.8 [39.6, 56.0]
X-Ord(187)	50.8 [43.9, 57.8]	43.3 [36.4, 50.3]	1.1 [0.0, 2.7]

Dimension (N)	Qwen3-VL-8B-Thinking	InternVL3-8B	Video-ChatGPT 7B
Ord(105)	10.5 [4.8, 17.1]	5.7 [1.9, 10.5]	7.6 [2.9, 13.3]
T-Loc(587)	34.9 [31.0, 38.7]	32.0 [28.3, 35.8]	18.1 [15.0, 21.3]
Dur(312)	24.7 [19.9, 29.5]	20.2 [15.7, 24.7]	19.6 [15.4, 24.0]
Cond(278)	40.6 [34.9, 46.4]	31.7 [26.3, 37.4]	20.5 [15.8, 25.2]
Move(178)	30.3 [23.6, 37.1]	26.4 [20.2, 33.1]	14.0 [9.0, 19.1]

Table 14: Format-specific accuracy [95% item-bootstrap interval]. Ord-3/4 require a fully correct permutation.

Format (N)	GPT-5.5	Kimi-K2.6	Qwen2.5-VL-72B
MCQ(1355)	45.6 [42.9, 48.3]	44.8 [42.1, 47.5]	35.4 [32.8, 37.9]
SP(563)	70.2 [66.3, 73.9]	63.2 [59.3, 67.1]	50.8 [46.7, 54.9]
Numerical(300)	18.3 [14.0, 22.7]	19.3 [15.0, 24.0]	3.0 [1.3, 5.0]
Ord-3(52)	19.2 [9.6, 30.8]	23.1 [11.5, 34.6]	19.2 [9.6, 30.8]
Ord-4(53)	30.2 [18.9, 43.4]	22.6 [11.3, 34.0]	11.3 [3.8, 20.8]

Table 15: Format-specific accuracy [95% item-bootstrap interval]. Ord-3/4 require a fully correct permutation.

Format (N)	Qwen3-VL-8B-Thinking	InternVL3-8B	Video-ChatGPT 7B
MCQ(1355)	33.1 [30.6, 35.7]	28.5 [26.1, 30.9]	18.4 [16.3, 20.4]
SP(563)	43.2 [39.1, 47.2]	44.8 [40.7, 48.8]	24.7 [21.1, 28.2]
Numerical(300)	14.0 [10.3, 18.0]	18.7 [14.3, 23.3]	30.7 [25.3, 36.0]
Ord-3(52)	15.4 [5.8, 25.0]	9.6 [1.9, 19.2]	13.5 [5.8, 23.1]
Ord-4(53)	5.7 [0.0, 13.2]	1.9 [0.0, 5.7]	1.9 [0.0, 5.7]

Artifact scope. The final manifest contains 2,323 unique QA IDs linked to 1,951 video IDs. A viewer showing 712 Perception Test video records displays one source subset, not the full benchmark. The QA-level index and original answer manifest describe the same items. Construction code covers tracking/post-processing, event extraction, QA synthesis, filtering, and verification export; evaluation code covers answer parsing and scoring. These resources and the source-video conditions are documented alongside the original release.