KAIROS: A DATASET FOR FINE-GRAINED VIDEO-LANGUAGE MODELING OVER SPACE, TIME, AND DYNAMICS

Anonymous authors
Paper under double-blind review



Figure 1: KAIROS represents long-form videos as structured, time-resolved annotation streams. Each video is decomposed into shots, with each shot annotated by a reference-frame description, an in-shot differential chain, and a video-level entity bank that links recurring identities across shots. At 1 FPS, the annotations capture three axes of fine-grained video understanding: *Space*, *Time*, and *Dynamics*. KAIROS-Bench is derived from KAIROS by converting these structured annotations into temporally grounded questions, whose answers are tied to explicit evidence spans.

ABSTRACT

Many emerging video language modeling tasks require systems to move beyond clip-level abstraction and model visual content as it unfolds over extended time horizons. However, most existing video datasets rely on coarse or sparsely aligned supervision, which compresses temporal variation and limits the ability of models to learn reusable representations of continuous visual dynamics. We introduce KAIROS, a video dataset for video-language modeling with time-resolved annotations. KAIROS consists of long-duration videos, ranging from ten minutes to half an hour, annotated with fine-grained temporal alignment. The annotations capture ongoing actions, entity appearances and attributes, interactions, and evolving contextual cues along the video timeline. This time-resolved structure supports fine-grained evaluation, long-range modeling and reasoning, instruction data construction, representation learning, and video generation. KAIROS provides a general-purpose foundation for modeling visual experiences over time.

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.

1 INTRODUCTION

Video-language modeling (Venugopalan et al., 2015; Sun et al., 2019; Zhu & Yang, 2020; Li et al., 2020; Luo et al., 2020; Fu et al., 2021; Li et al., 2022; Xu et al., 2021; Alayrac et al., 2022; Li et al., 2023; Zhang et al., 2023; Li et al., 2024a) seeks to represent and reason over video content through language as it unfolds over both space and time. In a video, entities appear, interact, and evolve continuously, forming structured patterns across both spatial layouts and temporal progressions. Such spatiotemporal evolution gives rise to rich dynamics, including event progression, state transitions, and causal dependencies, and provides the foundation for grounded reasoning about what happens, why it happens, and what may follow. Therefore, fine-grained video-language modeling requires jointly modeling Space, Time, and Dynamics: **Space** specifies what is present and how it is spatially arranged, **Time** captures when events occur across the video timeline, and **Dynamics** characterizes how entities, actions, and contexts evolve and interact to form coherent video-level understanding.

A fundamental barrier to achieving this goal lies in the data. Significant efforts have been made to advance video understanding, including construction of video caption datasets (Chen et al., 2024b; Farré et al., 2024), video instruction tuning and conversational video datasets (Luo et al., 2023; Zhang et al., 2024; Chen et al., 2024b; Ren et al., 2024), and recent long video benchmarks (Chen et al., 2024a; Qin et al., 2025; Mangalam et al., 2023; Wu et al., 2024; Cheng et al., 2025; Fu et al., 2025; Chandrasegaran et al., 2024; Li et al., 2024b; Wang et al., 2025a). Despite these advances, almost all prior works still fall short on all three fronts outlined above.

1. **Space:** Existing datasets still lack spatial granularity. Most rely on holistic video captions and place far less emphasis on fine-grained visual annotation than image datasets. As a result, their annotations typically foreground only the most salient objects, leaving many entities, attributes, and relations essential for a comprehensive account of video content unspecified.
2. **Time:** Existing datasets offer limited temporal resolution. Most provide annotations at the clip level, such as a single caption or a set of questions, without fine-grained temporal alignment. While such annotations capture high-level semantics, they fail to reflect how visual content, events, and scene states evolve over time.
3. **Dynamics:** Existing datasets, especially benchmarks, largely fail to probe fine-grained spatiotemporal grounding. This includes aspects such as pinpointing precisely when an event occurs, tracing how states change over time, and examining how observations from different moments jointly support reasoning about temporal order, interactions, and causality. As a result, they provide only limited supervision and evaluation of such dynamics, leaving models inadequately assessed in their ability to perform grounded reasoning over dynamic video structure.

These limitations are especially pronounced in long-form videos (e.g., longer than 10 minutes), where richer temporal dependencies accumulate over extended durations but remain sparsely annotated.

We propose KAIROS, a new dataset for fine-grained video-language modeling over Space, Time, and Dynamics. The core of KAIROS lies an automated annotation pipeline that produces fine-grained spatiotemporal descriptions at multiple levels of granularity: within each shot, it records detailed visual content; across frames within a shot, it captures temporal variation and scene evolution; and over the full video, it integrates these observations into spatially detailed and temporally resolved descriptions. A key component of this framework is a unified, text-centric representation that maintains consistency across long video contexts. KAIROS integrates entity consistency directly into frame-level annotations, allowing recurring entities and semantic attributes to be resolved over time. This design links entities across distant segments through language, while incorporating multimodal signals such as speech and environmental audio into a coherent narrative.

As demonstrated in Figure 1, built on this pipeline, KAIROS provides rich, time-resolved annotations for constructing both training data and benchmarks for video-language models. Since evidence may be distributed across distant temporal segments, KAIROS supports tasks such as cross-temporal reference resolution, causal reasoning, and compositional reasoning. Its annotations can be converted into dense dynamic captions, question-answer pairs, and explicit reasoning paths grounded in temporally localized evidence. This enables questions about what happened, when an entity appeared, how entities interacted, and how a scene evolved over time, together with rationales that connect answers to the corresponding video evidence.

Our final dataset contains 19,004 videos with an average duration of 17 minutes, totaling 5,420 hours of annotated video across 202 diverse scenario types. We focus on videos ranging from 10 to 30 minutes in length to better reflect real-world temporal horizons while maintaining high annotation quality. This regime is particularly important because the limitations of existing datasets become even more severe in long-form videos, where richer temporal dependencies accumulate over extended durations while annotations remain sparse. From a subset of 820 videos, we further construct a benchmark of 3,011 questions, all of which are passed through sanity checks and human verification. These questions are temporally grounded to specific moments in the video and include explicit reasoning rationales. As illustrated in Figure 1, KAIROS exhibits a substantial leap compared to caption-based datasets in annotation density and information richness.

Our extensive experiments on state-of-the-art video language models (OpenAI, 2026c; The Gemini Team, 2026; Intelligence, 2024; ByteDance Seed, 2026; Chen et al., 2024c; Bai et al., 2025; Xiaomi, 2025) reveal that performance degrades when the evidence spans several shots or the whole video. Besides, we highlight that fine-tuning on training data derived from KAIROS can improve the performance of video-language models on other long video benchmarks.

Our contributions are threefold.

1. We introduce KAIROS, a new long-form video-language dataset designed for fine-grained modeling of Space, Time, and Dynamics.
2. We develop an automated annotation pipeline that produces multi-level spatiotemporal representations for long videos, which captures detailed and structured visual content and enabling coherent tracking of recurring entities and semantic attributes over time.
3. We construct a temporally grounded benchmark and demonstrate the utility of KAIROS for evaluating and improving video-language models.

2 THE KAIROS DATASET

Annotation format. KAIROS represents each video as a dense, temporally unfolding annotation stream. Rather than assigning a single coarse clip-level description, it refreshes annotations at 1 FPS to capture fine-grained visual, auditory, and semantic changes as they occur. The format also preserves video structure: frames are organized into locally coherent shots, while a video-level entity matching mechanism tracks identities, attributes, and relationships as they persist, disappear, reappear, or are disambiguated across shots. Formally, each extracted frame is stored as a structured record with four fields: **(1)** an absolute timestamp, **(2)** the corresponding video frame, **(3)** a shot index, and **(4)** an audio-aware description. Together, these records provide a temporally grounded interface for fine-grained video understanding, retrieval, temporal reasoning, entity tracking, and video generation.

2.1 KAIROS ANNOTATION PIPELINE

The dense and structured nature of KAIROS annotations makes manual annotation prohibitively expensive. We therefore develop an automated pipeline that first structures the video globally and then performs fine-grained annotation. As shown in Figure 2, this design preserves both local details and long-range video dynamics under a tractable computational budget.

Video structure parsing. We model cross-shot entity consistency to handle entities that reappear across distant shots, undergo visual changes, or are later identified through speech and context, thereby capturing continuity and long-range relationships across the video. We first employ `TransNetV2` (Soucek & Lokoc, 2024) to segment each video into shots. After shot segmentation, we identify entities appearing in the reference frame of each shot and construct a video-level entity bank. We define eight types of entities: `person`, `animal`, `object`, `vehicle`, `text`, `location`, `food`, and `clothing`. Each detected entity is associated with a canonical label and a short visual-detail string, describing distinguishing attributes. We then link each entity mention emitted for a new reference frame to the video-level entity bank. The bank is injected into every reference-frame prompt, which requires that an entity already in the bank be referred to by its canonical name verbatim; a mention whose normalized label exactly matches a bank entry is recorded as a new appearance of that entity. This lets later shots reuse established identities instead of introducing

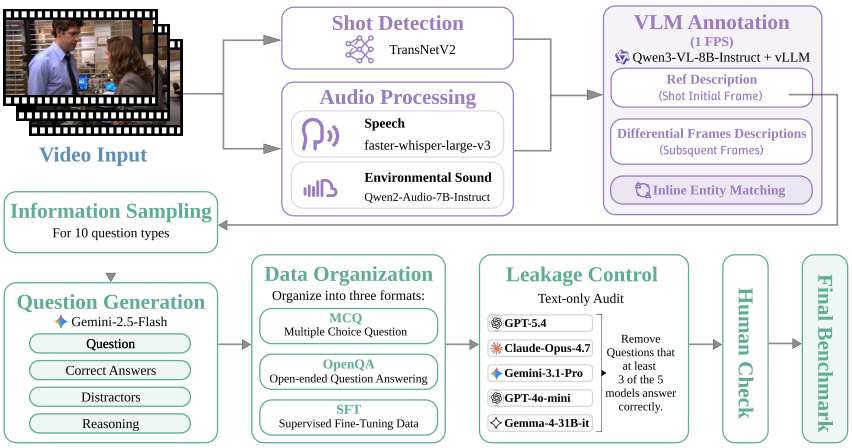

Figure 2: KAIROS transforms raw long-form videos into structured, time-resolved annotations that integrate visual, temporal, entity-level, and audio information, supporting dense captioning, reasoning supervision, and KAIROS-Bench construction.

duplicates. In this way, the entity bank serves both as a tracking mechanism and as global context for spatio-temporal descriptions.

In-shot fine-grained annotation. Given the parsed video structure, KAIROS annotates each shot at 1 FPS with an initial-and-subsequent scheme. The first sampled frame in each shot is treated as the *reference frame* and receives a full moment-level description, including spatial layout, visible entities, attributes, relations, on-screen text, and relevant audio cues. Subsequent 1 FPS samples are treated as *differential frames*, which record only fine-grained changes relative to the previous second, such as actions, state transitions, motion, interactions, visibility changes, and newly appearing or disappearing entities. This design reduces redundancy while preserving fine-grained temporal evolution. The core visual annotation process is driven by `Qwen3-VL-8B-Instruct` (Bai et al., 2025) and accelerated with `vLLM` (Kwon et al., 2023). For each initial or differential frame, the model receives the sampled frame, the current entity bank, and the relevant context history and audio information. The generated description is structured along three axes: *Space*, describing what is present at a moment; *Time*, anchoring the description to an absolute point on the video timeline; and *Dynamics*, describing how the scene evolves.

Audio and transition annotation. We combine speech transcription and non-speech audio summarization. `faster-whisper-large-v3` (SYSTRAN, 2024; Radford et al., 2023) transcribes speech into timestamped sentence-level segments, while `Qwen2-Audio-7B-Instruct` (Chu et al., 2024) summarizes ambient and event-level sounds in 30-second windows. Both streams are aligned with the annotation timeline: speech is attached to the corresponding frame, while non-speech summaries are attached only to reference frames to preserve context without redundancy. KAIROS also explicitly annotates shot-boundary transitions to preserve cross-shot continuity. For each boundary, the pipeline records the editing technique (hard cut, fade, dissolve, wipe, etc.), the narrative purpose of the cut, and the visual contrast across it. These descriptions link adjacent shots and prevent the video annotation from becoming a set of disconnected shot-level records.

Annotation computation analysis. Because KAIROS annotations are produced by a fully automated pipeline, the process scales naturally with available GPU parallelism. In our production setting, one H200 annotates about 4 hours of video per hour; with tensor parallelism over two H200s, the pipeline reaches about $8\times$ real time. This corresponds to roughly 1,355 H200-hours for the KAIROS dataset of 5,420 hours, with preprocessing and audio analysis running at over $20\times$ real time and contributing little to the overall cost.

2.2 VIDEO CURATION

To provide a suitable setting for studying dynamic and fine-grained video understanding, we explicitly prioritize long-form narratives with extended durations and a high density of shot transitions. Such videos naturally encapsulate rich event progressions, entity interactions, and causal dependencies over time. We source our raw data from the two most prominent long-form video platforms:

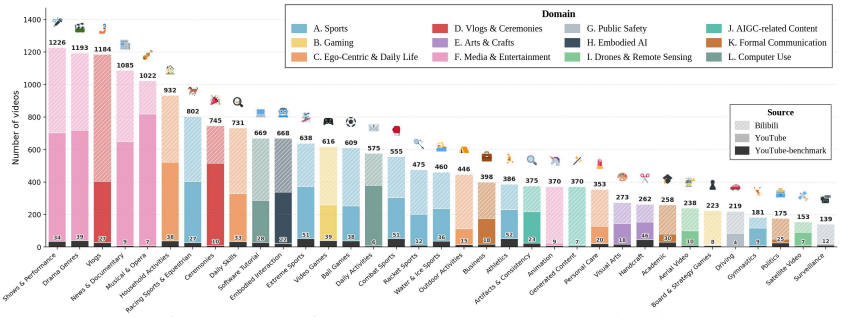

Figure 3: Per-category video count of KAIROS dataset, grouped by the 12 parent domains (bar color). Each bar stacks the curated benchmark subset (820 videos, dark bottom), the YouTube non-benchmark remainder (middle), and the Bilibili (top hatched).

YouTube and *Bilibili*. Using an agent-based search strategy, we constructed an initial pool of candidate videos ranging from 10 minutes to 4 hours in length. This yielded a total of 28,282 URLs, comprising 15,054 from YouTube and 13,228 from Bilibili.

To ensure semantic diversity, each candidate URL is pre-annotated and filtered using a stringent three-level taxonomy encompassing 12 high-level domains, 35 categories, and 202 leaf-level scenarios (e.g., *A. Sports* $\rightarrow$ *A.1. Ball Games* $\rightarrow$ *A.1.1. Basketball*). During processing, excessively long videos are segmented into consecutive clips of no more than 30 minutes to maintain annotation fidelity while preserving long-range context. Our final annotated dataset consists of 19,004 high-quality videos, of which 9,434 are from YouTube, and 9,570 are from Bilibili. The total is 5,420 hours. As illustrated in Figure 3, KAIROS exhibits substantial diversity.

3 THE KAIROS-BENCH

The dense, time-resolved annotations of KAIROS provide structured evidence for downstream video-language tasks. We use them to construct KAIROS-Bench, a benchmark for fine-grained video understanding, by guiding LLM-based question generation with a taxonomy of capabilities and temporal tiers. Each item is derived from the relevant evidence in the annotation stream and produced as a complete multiple-choice instance, including the question, answer, distractors, and rationale. The same mechanism can also be adapted to generate instruction-response pairs for downstream training. Because benchmark construction requires a higher quality standard than raw generation, we further apply a strict verification process before including questions in KAIROS-Bench.

3.1 BENCHMARK DESIGN

We design the KAIROS-Bench around three orthogonal axes that are often entangled in long-form video understanding: *Space*, *Time*, and *Dynamics*. The *Space* axis measures what is present at a single moment, including scenes, entities, spatial relations, on-screen text, and audio cues. The *Time* axis measures where the required evidence resides and how long the evidence span is, ranging from moment-level cues to whole-video context. The *Dynamics* axis measures how the video evolves, including within-shot changes, cross-shot continuity, temporal ordering, causal relations, counterfactual reasoning, counting, and holistic integration. To systematically probe these dimensions, we partition questions into 17 capabilities across four cognitive levels: second-level perception, intra-shot evolution, cross-shot reasoning, and whole-video understanding. Each question is therefore associated with both a capability label and a temporal tier, allowing model performance to be analyzed at a finer granularity.

Given a video, the generator samples evidence only from the timestamped KAIROS annotation stream, and asks `Gemini-2.5-Flash` (Comanici et al., 2025) to produce a complete multiple-choice question (MCQ) in a single call. This design makes question construction scalable while keeping every question grounded in the same fine-grained evidence used by the dataset.

We design ten source-type samplers, each targeting a different temporal scale and semantic structure in the annotation stream:

- **Reference-frame samplers.** The `ref` samplers target local evidence from reference frames. `ref_perception` samples scene, entity, and spatial information, `ref_ocr` samples on-screen text, and `ref_audio` samples speech or environmental sound.
- **Within-shot samplers.** The `diff` samplers target short-range dynamics within a shot. `diff_change` samples a reference frame with several subsequent differential descriptions, while `diff_sequence` samples the full differential chain of a shot.
- **Cross-shot samplers.** `entity_tracking`, `transition`, `cross_shot`, and `long_range` sample entities, events, and transitions across multiple shots, from adjacent-shot changes to multi-minute evidence windows.
- **Full-video sampler.** `full_video` samples evidence over an entire video or a long segment, supporting holistic questions that require extended temporal integration.

Each sampler returns the materials needed for question generation, including the relevant annotation descriptions, their timestamps, the evidence span, neighboring reference-frame context, and a pool of candidate distractors. The evidence span determines the temporal tier of the question: `T1` for moment-level evidence, `T2` for evidence within 60 seconds, `T3` for evidence within 300 seconds, `T4` for evidence within 900 seconds, and `T5` for evidence beyond 900 seconds. At the same time, the source type restricts the possible capability labels from the Space and Dynamics taxonomy. Thus, the temporal tier and capability label are not assigned post hoc after question generation; they are determined by the evidence sampling process itself.

In the second step, we provide `Gemini-2.5-Flash` (Comanici et al., 2025) with the sampled evidence, the neighboring reference-frame context before and after the target evidence, and a coarse temporal hint indicating the approximate position of the evidence in the video. The model is required to return a structured response containing four fields: `question`, `answer`, `distractors`, and `reasoning`. The question, correct answer, three distractors, and rationale are generated atomically in the same call. This encourages internal consistency between the answer and the reasoning, and avoids a separate rewriting stage that could make the distractors superficially different from the correct answer.

The distractors design. A key design choice is to construct distractors from real annotations rather than hallucinating from scratch. Specifically, the distractor pool is drawn from non-overlapping time windows of the same video. Thus, wrong options remain linguistically and semantically plausible because they describe content that actually appears in the video, but they refer to the wrong time point. This reduces text leakage and prevents models from relying on commonsense priors or eliminating obviously implausible choices. To answer correctly, a model must locate the relevant temporal evidence and understand the corresponding visual, auditory, or dynamic content.

3.2 Checks of the Benchmark

Leakage control and sanity checks. A common failure mode in video benchmarks is text leakage, where a model can infer the correct answer from the question wording or option priors alone, without actually understanding the video. We address this issue at both the generation and filtering stages. At the generation stage, distractors are designed to be plausible rather than fabricated. This prevents models from answering by simply eliminating obviously implausible choices, instead forces them to locate and understand the relevant visual, auditory, and dynamic evidence. With this design, some questions may still be solvable from textual priors. We therefore apply a strict text-only audit using a committee of five strong language models: `GPT-5.4` (OpenAI, 2026b), `Claude-Opus-4.7` (Anthropic, 2026), `Gemini-3.1-Pro` (The Gemini Team, 2026), `GPT-4o-mini` (OpenAI, 2026a), and `Gemma-4-31B-it` (Google DeepMind, 2026). Each model receives only the question and the four shuffled answer options, without access to the video frames, audio, or textual video annotations. If at least three of the five models answer a question correctly, the question is marked as text-solvable and removed. Since each model is evaluated with a single shuffled option order, the random majority floor is approximately 10.4%. Across 26,363 raw questions, this text-only audit removes 19,920 items and leaves 6,443 video-dependent candidates.

Human review. After the text-only audit, we further introduce a human verification stage to construct the final curated evaluation set. Each surviving MCQ is independently reviewed by at least two human annotators, who watch the corresponding video and evaluate the item according to four criteria. First, **question validity** checks whether the question is clear, unambiguous, and answerable given the video evidence. Second, **accuracy and grounding** verifies that the correct answer

Table 1: Using a text-only `Gemini-3.1-Pro` solver with identical prompts and option shuffling, KAIROS-Bench shows the lowest leakage among all benchmarks, measured by absolute lift over the random baseline, despite having the second-longest question stems.

Benchmark	# Options	Random	Accuracy	Leakage ↓	# Avg words ↑
CG-Bench (Chen et al., 2024a)	var 2–8, $\bar{N}$=6.86	14.57%	50.20%	+35.63 pp	48
DeVE-QA (Qin et al., 2025)	fixed 5	20.00%	53.60%	+33.60 pp	37
EgoSchema (Mangalam et al., 2023)	fixed 5	20.00%	53.60%	+33.60 pp	**154**
LongVideoBench (Wu et al., 2024)	fixed 4	25.00%	56.00%	+31.00 pp	87
Video-Holmes (Cheng et al., 2025)	fixed 6	16.67%	47.60%	+30.93 pp	54
Video-MME (Fu et al., 2025)	fixed 4	25.00%	54.80%	+29.80 pp	40
HourVideo (Chandrasegaran et al., 2024)	fixed 5	20.00%	39.00%	+19.00 pp	97
MVBench (Li et al., 2024b)	var 2–5, $\bar{N}$=3.51	28.45%	41.00%	+12.55 pp	32
LVBench (Wang et al., 2025a)	fixed 4	25.00%	37.40%	+12.40 pp	39
KAIROS (Ours)	fixed 4	25.00%	34.60%	**+9.60 pp**	144

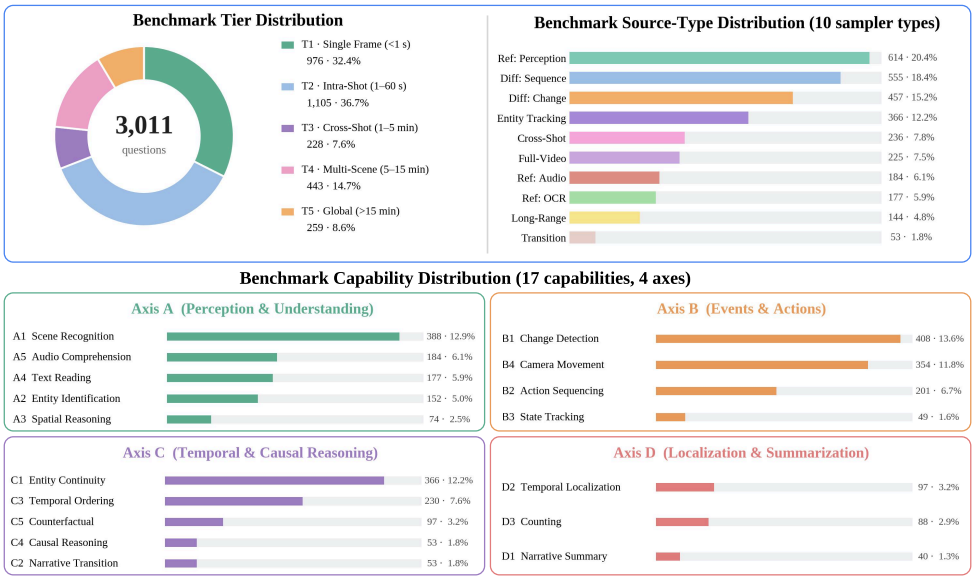

Figure 4: Statistics of KAIROS-Bench. Distribution of the curated 3,011 questions across three independent labeling axes: evidence span, source type, and evaluated capability, showing the benchmark coverage over space, time, and dynamics.

matches the visual or auditory facts and that the associated temporal evidence is accurate. Third, **answer uniqueness** ensures that none of the distractors can reasonably be interpreted as another correct answer. Fourth, **video dependence** confirms that the question cannot be answered easily without watching the video. Any item that fails the review criteria is discarded. For the final leaderboard set, we retain only questions with unanimous pass verdicts, yielding 3,011 human-curated MCQs over 820 videos. As shown in Table 1, we keep the text-only accuracy of KAIROS-Bench substantially lower than that of common video benchmarks, indicating that the retained questions require video-grounded evidence rather than language-only reasoning.

3.3 BENCHMARK STATISTICS

Following the generation, leakage-control, and human-review steps above, the released KAIROS-Bench contains 3,011 human-verified MCQs (and matched Open-ended QA pairs) over 820 videos drawn from all 35 categories. Figure 4 summarizes the three labeling axes: the *Time* axis is deliberately weighted toward the long-context tail (T3–T5 jointly account for ≈30% of items), the *Source* axis is dominated by reference-frame and within-shot samplers but retains meaningful mass on cross-shot and full-video samplers, and the *Capability* axis spreads over all 17 rows so that per-axis evaluation surfaces specific failure modes rather than a single overall score. Representative questions across these axes are shown in Figure 5.

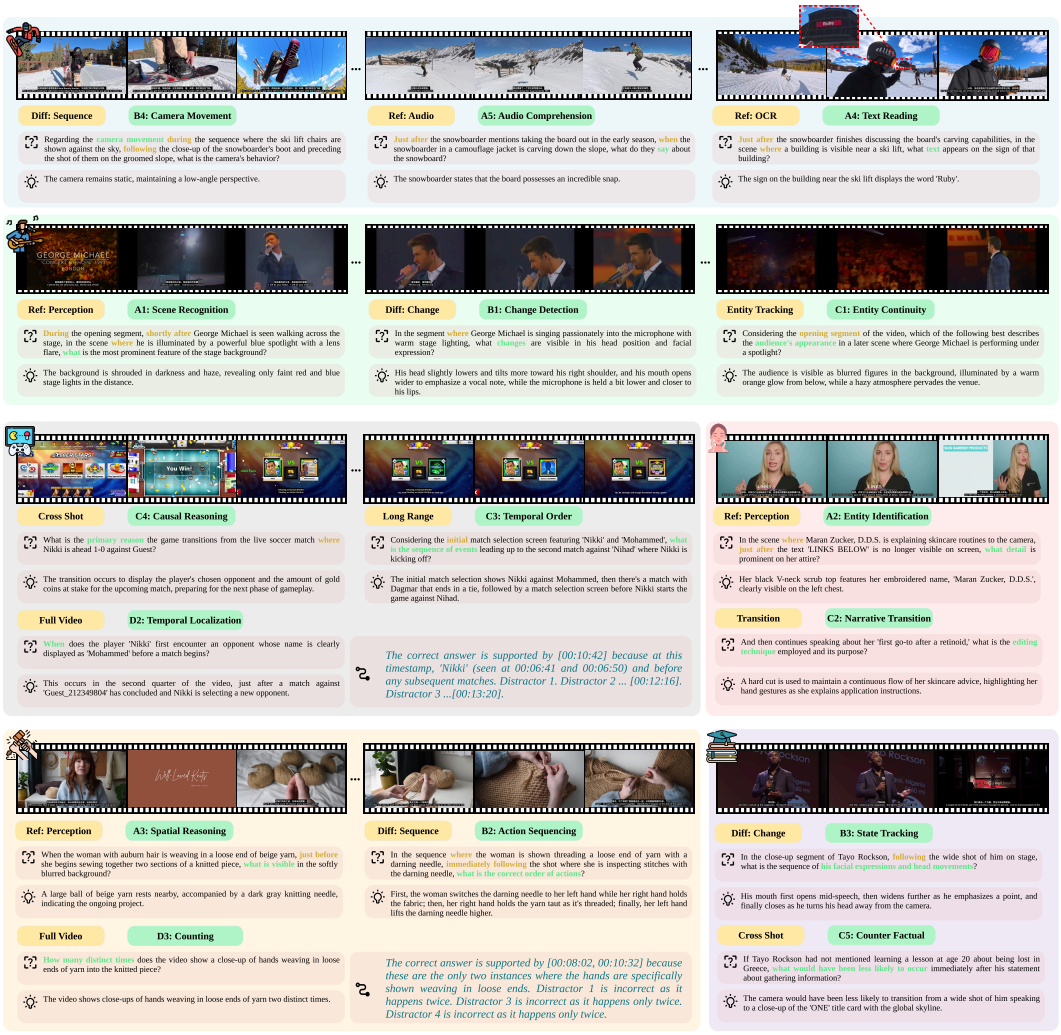

Figure 5: Representative questions from KAIROS-Bench across the three axes. Each block shows three keyframes, the question and correct answer, and the labels assigned to the question. The examples cover the full evidence span: from single-frame perception, through within-shot evolution and cross-shot continuity, up to full-video.

3.4 RESULTS OF THE KAIROS-BENCH

We evaluate 7 closed-source models and 14 open-source models. Three models receive the video natively: the two `Gemini` models and `LLaVA-Video-7B`. Every other model receives uniformly sampled frames at the per-model budget listed in Table 8. The results are shown in Table 2. Accuracy drops once the evidence spans several shots or the whole video.

To further analyze text leakage, we evaluate public video-MCQ benchmarks under an identical protocol: we draw a stratified sample of 500 questions from each benchmark and answer them with `Gemini-3.1-Pro` (The Gemini Team, 2026) using text only, without any visual input. We define leakage as the solver accuracy minus the random-answering baseline for each benchmark. KAIROS-Bench exhibits the lowest leakage despite having the second-highest word count. This is notable because KAIROS-Bench intentionally avoids exposing numeric timestamps that would allow a model to directly retrieve or attend to the referenced segment. Instead, each question contains a linguistic anchor that pins the query to a specific moment in the video while still requiring the model to localize the relevant event from visual content. This design prevents models from taking a timestamp-based shortcut, yet it also makes text-only leakage a more serious concern because the questions must include richer natural-language grounding. To mitigate this risk, KAIROS-Bench undergoes a strict auditing process involving five models followed by human verification.

Table 2: MCQ leaderboard on the KAIROS-Bench with full per-capability accuracy. The 17 capabilities are grouped by their temporal scope. *#Q* is the question count per capability.

Model	Overall	Single Frame					Within Shot				Cross Shots					Full Video		
		A1	A2	A3	A4	A5	B1	B2	B3	B4	C1	C2	C3	C4	C5	D1	D2	D3
#Q	3,011	388	152	74	177	184	408	201	49	354	366	53	230	53	97	40	97	88
Closed-source																		
`Gemini-3.1-Pro` (The Gemini Team, 2026)	**58.1**	**61.1**	54.6	67.6	**72.3**	**66.8**	53.7	42.8	44.9	54.8	**64.5**	26.4	**61.3**	**69.8**	46.4	75.0	**81.4**	27.3
`GPT-5.5` (OpenAI, 2026c)	56.7	57.1	57.4	**78.3**	64.5	39.9	55.6	46.9	**51.2**	**69.5**	58.2	42.0	58.8	57.1	37.3	**81.6**	74.0	26.5
`Gemini-2.5-Flash` (Comanici et al., 2025)	53.4	53.4	**60.5**	74.3	66.1	53.8	**56.4**	**47.3**	44.9	54.2	57.1	17.0	47.0	56.6	32.0	60.0	52.6	**40.9**
`Nova-2-Lite` (Intelligence, 2024)	46.4	47.7	47.4	59.5	48.6	37.0	55.9	40.3	42.9	47.2	42.9	**49.1**	43.9	60.4	**48.5**	50.0	39.2	28.4
`Seed-2.0-Lite` (ByteDance Seed, 2026)	42.3	50.8	48.0	58.1	47.5	31.5	41.7	35.8	46.9	40.4	41.8	22.6	37.4	39.6	35.1	65.0	48.5	37.5
`GPT-4o` (Achiam et al., 2023)	38.2	42.8	43.4	50.0	44.1	31.0	32.4	32.8	28.6	39.5	42.4	28.3	33.5	34.0	28.9	75.0	42.3	34.1
`GPT-4o-mini` (OpenAI, 2026a)	31.5	32.7	27.6	32.4	32.2	23.9	35.0	29.9	38.8	24.9	35.2	30.2	31.3	41.5	27.8	75.0	19.6	33.0
Open-weight																		
`InternVL3-78B` (Chen et al., 2024c)	**51.7**	**47.9**	**46.7**	**64.9**	53.7	**37.0**	**61.3**	**45.3**	**51.0**	62.4	**50.3**	**47.2**	**46.1**	**64.2**	44.3	75.0	47.4	39.8
`InternVL3.5-38B` (Wang et al., 2025b)	47.0	46.9	46.7	62.2	50.3	29.9	53.4	35.3	34.7	54.5	42.1	47.2	41.3	62.3	**52.6**	75.0	**48.5**	**44.3**
`GLM-4.5V` (Hong et al., 2025)	46.5	44.8	44.1	54.1	53.7	34.8	50.0	39.8	49.0	**65.8**	41.3	37.7	37.4	45.3	43.3	62.5	47.4	28.4
`Qwen3-VL-30B-A3B` (Bai et al., 2025)	45.5	43.3	42.8	58.1	**54.2**	28.8	46.8	37.3	42.9	63.8	42.4	39.6	38.7	54.7	41.2	**77.5**	41.2	30.7
`Qwen3-VL-8B` (Bai et al., 2025)	42.3	42.0	43.4	54.0	52.5	26.1	42.6	38.3	40.8	50.6	42.1	34.0	34.4	52.8	37.1	67.5	39.2	38.6
`MiMo-VL-7B` (Xiaomi, 2025)	41.9	41.5	41.4	54.1	47.5	24.5	42.2	37.8	30.6	55.1	41.8	30.2	40.4	54.7	37.1	60.0	33.0	33.0
`InternVL3-8B` (Zhu et al., 2025)	41.6	46.1	36.2	55.4	45.8	30.4	51.7	38.8	36.7	35.3	37.4	37.7	39.1	54.7	40.2	60.0	36.1	39.8
`Qwen2.5-VL-7B` (Bai et al., 2023)	41.1	37.6	40.8	54.0	44.1	35.3	42.4	39.3	40.8	48.6	38.0	39.6	37.8	45.3	43.3	57.5	36.1	37.5
`InternVL3.5-8B` (Wang et al., 2025b)	40.7	42.0	36.8	43.2	47.5	29.3	44.6	37.8	36.7	44.1	41.3	41.5	39.6	52.8	25.8	62.5	33.0	34.1
`Step3-VL-10B` (Team, 2025)	39.3	41.5	40.1	50.0	53.7	23.9	38.7	29.9	32.7	45.2	41.5	22.6	31.7	49.1	38.1	65.0	39.2	31.8
`GLM-4V-9B` (GLM et al., 2024)	38.7	39.7	36.2	37.8	35.0	27.7	41.2	33.3	49.0	46.6	39.9	45.3	36.5	47.2	40.2	50.0	25.8	31.8
`Gemma-4-31B` (Google DeepMind, 2026)	38.1	42.3	46.7	58.1	42.9	26.1	33.1	26.9	36.7	42.1	39.3	34.0	34.8	41.5	38.1	50.0	41.2	33.0
`CogVLM2-Video-13B` (Hong et al., 2024)	36.2	33.8	36.8	36.5	31.1	29.4	40.2	33.3	40.8	49.4	36.9	26.4	30.9	39.6	32.0	55.0	22.7	28.4
`LLaVA-Video-7B` (Li et al., 2024a)	26.8	33.0	16.4	33.8	24.3	28.8	22.1	22.4	26.5	28.0	31.1	28.3	27.8	32.1	17.5	37.5	21.6	27.3

Table 3: OpenQA leaderboard on the KAIROS-Bench. `Gemini-2.5-Flash` (Comanici et al., 2025) judge scores each answer against the ground truth on a 0-3 scale, and we report per-tier binarized accuracy with scores no less than 2 counted as correct.

Model	Overall	T1	T2	T3	T4	T5
#Q	3,011	976	1,105	228	443	259
`MiMo-VL-7B` (Xiaomi, 2025)	**22.6%**	**27.3%**	**19.4%**	**21.9%**	19.6%	24.3%
`GLM-4.5V` (Hong et al., 2025)	22.5%	27.2%	18.1%	20.2%	**23.3%**	23.9%
`Qwen3-VL-30B-A3B` (Bai et al., 2025)	22.5%	27.2%	18.2%	16.2%	22.1%	**29.0%**
`InternVL3-78B` (Chen et al., 2024c)	19.7%	26.3%	13.7%	17.5%	20.5%	20.5%
`Step3-VL-10B` (Team, 2025)	19.5%	25.9%	15.2%	14.0%	18.7%	19.7%
`Qwen3-VL-8B` (Bai et al., 2025)	19.2%	25.9%	12.0%	14.0%	22.8%	22.8%
`Qwen2.5-VL-7B` (Bai et al., 2023)	19.0%	22.7%	14.9%	15.4%	19.0%	25.9%
`InternVL3.5-38B` (Wang et al., 2025b)	16.6%	22.7%	11.2%	15.8%	18.5%	14.3%
`InternVL3.5-8B` (Wang et al., 2025b)	14.9%	20.8%	9.5%	13.2%	15.6%	16.6%
`InternVL3-8B` (Zhu et al., 2025)	14.9%	21.0%	9.2%	13.2%	16.0%	16.2%
`GLM-4V-9B` (GLM et al., 2024)	10.6%	16.3%	6.4%	8.3%	8.8%	11.6%
`LLaVA-Video-7B` (Li et al., 2024a)	8.6%	9.6%	3.3%	6.2%	15.0%	19.3%

Table 4: KAIROS training data improves `Qwen2.5-VL-7B-Instruct` (Bai et al., 2023) on KAIROS-Bench and three external long-video benchmarks via LoRA (Hu et al., 2022) fine-tuning. At evaluation stage, we ablate the frame budget over $\{16, 32, 64\}$. Without external training data, the fine-tuned model improves over the base across all benchmarks.

Model	Benchmark			
	KAIROS	LongVideoBench	LVBench	Video-MME
`base` (32-f)	40.42	56.29	38.69	57.11
`finetuned` (16-f)	**47.94**	58.00	39.18	57.63
`finetuned` (32-f)	47.49	**60.08**	40.81	**59.56**
`finetuned` (64-f)	45.12	59.50	**42.15**	59.30

We additionally run an OpenQA pass on open-source models on KAIROS-Bench: the model must generate the answer rather than pick a letter. We score with a text-only LLM judge `Gemini-2.5-Flash`, which gets only question, reference answer, and candidate. It returns an integer 0–3 (no less than 2 counts as correct). The results are shown in Table 3.

3.5 EMPOWERING VIDEO-LANGUAGE MODELS WITH KAIROS

We fine-tune `Qwen2.5-VL-7B-Instruct` with LoRA for 1 epoch, on 232,101 SFT samples (each with the question, answer and reasoning) derived from the KAIROS training split. The model is trained using uniformly sampled 32 frames per video. We evaluate on KAIROS-Bench and three public long-video benchmarks (LongVideoBench, LVBench, and Video-MME). Without external training data, the fine-tuned model improves over the base across all benchmarks. The results are shown in Table 4, proving that KAIROS serves as a valid supervision target.

4 CONCLUSION

We present KAIROS, a dataset of 19,004 long-form videos with 1 FPS temporally grounded annotations, and KAIROS-Bench, a strictly audited benchmark of 3,011 MCQ and OpenQA questions organized along three orthogonal axes (Space, Time, Dynamics). KAIROS-Bench is the cleanest of nine public video-MCQ benchmarks under an identical text-only-leakage probe. Fine-tuning a 7B open-weight VLM on 232,101 SFT samples derived from the KAIROS dataset improves accuracy on KAIROS and on three external long-video benchmarks despite using no training data from them. We release the dataset, benchmark, and pipeline to enable evaluation and supervision of long-form video understanding at the granularity at which it actually unfolds.

AI USE STATEMENT

Generative models are not an authoring aid in this work but a component of the artifact itself, and every such use is documented in the main text and the appendix. The annotation pipeline is built on generative models: `Qwen3-VL-8B-Instruct` (Bai et al., 2025) writes the per-frame reference, differential, and transition descriptions and emits the entity records (§B), `faster-whisper-large-v3` (SYSTRAN, 2024; Radford et al., 2023) transcribes speech, and `Qwen2-Audio-7B-Instruct` (Chu et al., 2024) summarizes non-speech audio. KAIROS-Bench questions, answers, distractors, and rationales are generated by `Gemini-2.5-Flash` (Comanici et al., 2025) from the annotation stream (§D), filtered by a five-model text-only audit, and scored in the open-ended setting by an LLM judge (§E). Because these outputs are the contribution rather than a convenience, they are validated rather than trusted: a stratified human audit of the annotation corpus (§G), a comparison against three larger annotation models (§H), component ablations (§I), and human verification of every released benchmark question against its source video (§J). The released annotations are model-generated; the model behind each annotation type is documented in §B and §D.

In preparing the manuscript, we used large language models for editing assistance only: grammar, spelling, and word choice on text written by the authors. We also used AI coding assistants while developing the pipeline and the evaluation scripts; all such code was read, tested, and corrected by the authors, and every reported number is produced by code that the authors verified. We did not use generative AI to originate research ideas, design the experiments, interpret the results, retrieve or generate citations, or write the scientific claims of this paper. We have reviewed all AI-assisted work and take full responsibility for the content of this paper, including its text, its claims, and the released artifacts.

ETHICS STATEMENT

Source material and licensing. KAIROS is derived from videos that were publicly available on two large video platforms. We do not redistribute the pixel data of those videos: the public release consists of the source URLs and our text annotations under CC BY 4.0, and the annotation pipeline is released so that the annotations can be regenerated from the sources (§L). Platform and creator ownership is respected: if a creator or a platform removes a video, or a rights holder submits a request, the corresponding URL and every record derived from it are removed from the release.

Human annotation. The audit of the annotation corpus and the verification of the benchmark questions were carried out by trained annotators recruited for this project. The task is a quality judgement about video annotations; no personal data about the annotators is released. The work involves no experimentation on human subjects and no intervention, so it did not require ethical review board approval under the applicable institutional policy.

Privacy and identifiable individuals. Long-form web video contains recognizable people, and our annotations describe what is visible in the frame. The pipeline adds no identifying information that the source does not already expose: it performs no face recognition, no biometric matching, and no linkage against external identity databases, and the cross-shot entity mechanism operates purely on the language of the descriptions (§B). Names appear only where they are stated in the video itself, for example in on-screen text or commentary.

Representation and bias. The corpus follows the distribution of public long-form web video on two platforms rather than a balanced geographic, linguistic, or demographic design, and the annotations inherit the perceptual biases of the models that produced them (§M). We document these limitations so that downstream users can account for them, and we report the platform and content-category composition of the corpus (§C).

Intended use and misuse. KAIROS is intended for training and evaluating long-form video understanding. Dense, time-resolved descriptions of people in public video could in principle support surveillance-style applications; we consider such use out of scope and inconsistent with the intended use stated in §M, and the release contains no biometric features, no identity linkage against external databases, and no video pixels; personal names and handles appear only where the source video itself states or displays them (e.g. athlete names in broadcast graphics).

Conflicts of interest. The authors declare no conflicts of interest or sponsorship that could have influenced the reported results.

REPRODUCIBILITY STATEMENT

Every stage of the work is documented at the level needed to rerun it. The annotation pipeline is specified in §B, which gives the shot detector settings, the frame sampling and encoding parameters, the four prompt templates, the cross-shot entity matching rule, the batching and engine parameters, and the models with their decoding settings; §C specifies the per-video output schema, the source curation, and the full content taxonomy. Benchmark construction is specified in §D: the ten source-type samplers, the mapping from evidence span to temporal tier, the atomic generation call, and the composition of the released set; the leakage protocol, including the solver, sample size, prompt, and random baseline, is given in §D.2. The evaluation protocol is specified in §E, with the backend, tensor-parallel size, frame budget, and decoding cap for every model in the leaderboard, and the answer-parsing and scoring rules; the fine-tuning recipe, including every hyperparameter and the visual input configuration, is in §F. The protocols behind the supporting analyses are given with them: the audit sampling and rubric in §G, the paired ablation design and its control in §I, the review statistics in §J, and the significance testing and additional training arms in §K.

The supplementary material accompanying this submission contains the complete annotation, benchmark-generation, training, and evaluation code, the benchmark in both its multiple-choice and open-ended forms, and, for the 820 benchmark videos, the source URLs and the complete text annotations, so every procedure described above can be inspected and rerun directly from that archive. No pixel data is redistributed in any form. The full annotated corpus will be released publicly under the terms in §L. Two dependencies limit exact reproduction and are stated rather than worked around. Benchmark generation, part of the text-only audit, and the open-ended judge use proprietary API models whose versions can change over time, so a regenerated benchmark may differ in detail from the released one. The corpus is defined by source URLs, so videos removed by their creators or platforms cannot be recovered from the release; the annotations, which are the contribution, remain available regardless.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. *Advances in neural information processing systems*, 34:24206–24221, 2021.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *Proceedings of the IEEE international conference on computer vision*, pp. 5803–5812, 2017.

Anthropic. Introducing Claude Opus 4.7. `https://www.anthropic.com/news/claude-opus-4-7`, 2026. Accessed: 2026-05-07.

Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6836–6846, 2021.

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

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*, 2025.

Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1728–1738, 2021.

Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *Icml*, volume 2, pp. 4, 2021.

ByteDance Seed. Seed2.0. https://seed.bytedance.com/en/seed2, 2026. Accessed: 2026-05-07.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017.

Keshigeyan Chandrasegaran, Agrim Gupta, Lea M. Hadzic, Taran Kota, Jimming He, Cristóbal Eyzaguirre, Zane Durante, Manling Li, Jiajun Wu, and Li Fei-Fei. Hourvideo: 1-hour video-language understanding, 2024. URL https://arxiv.org/abs/2411.04998.

Guo Chen, Yicheng Liu, Yifei Huang, Yuping He, Baoqi Pei, Jilan Xu, Yali Wang, Tong Lu, and Limin Wang. Cg-bench: Clue-grounded question answering benchmark for long video understanding, 2024a. URL https://arxiv.org/abs/2412.12075.

Lin Chen, Xilin Wei, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Bin Lin, Zhenyu Tang, et al. Sharegpt4video: Improving video understanding and generation with better captions. *Advances in Neural Information Processing Systems*, 37:19472–19495, 2024b.

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 *CVPR*, 2024c.

Junhao Cheng, Yuying Ge, Teng Wang, Yixiao Ge, Jing Liao, and Ying Shan. Video-holmes: Can mllm think like holmes for complex video reasoning?, 2025. URL https://arxiv.org/abs/2505.21374.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, et al. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759*, 2024.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Han Fang, Pengfei Xiong, Luhui Xu, and Yu Chen. Clip2video: Mastering video-text retrieval via image clip. *arXiv preprint arXiv:2106.11097*, 2021.

Miquel Farré, Andi Marafioti, Lewis Tunstall, Leandro Von Werra, and Thomas Wolf. Finevideo. https://huggingface.co/datasets/HuggingFaceFV/finevideo, 2024.

Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6202–6211, 2019.

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.

Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. Violet: End-to-end video-language transformers with masked visual-token modeling. *arXiv preprint arXiv:2111.12681*, 2021.

Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *Proceedings of the IEEE international conference on computer vision*, pp. 5267–5275, 2017.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, Lei Zhao, Lindong Wu, Lucen Zhong, Mingdao Liu, Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao, Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiao Xia, Xiaohan Zhang, Xiaotao Gu, Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yifan An, Yifan Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang, Zhen Yang, Zhengxiao Du, Zhenyu Hou, and Zihan Wang. Chatglm: A family of large language models from glm-130b to glm-4 all tools, 2024.

Google DeepMind. Gemma 4 Model Card. `https://ai.google.dev/gemma/docs/core/model_card_4`, 2026. Accessed: 2026-05-07.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18995–19012, 2022.

Wenyi Hong, Weihan Wang, Ming Ding, Wenmeng Yu, Qingsong Lv, Yan Wang, Yean Cheng, Shiyu Huang, Junhui Ji, Zhao Xue, et al. Cogvlm2: Visual language models for image and video understanding. *arXiv preprint arXiv:2408.16500*, 2024.

Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, et al. Glm-4.1 v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. pp. arXiv–2507, 2025.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=nZeVKeeFYf9`.

Amazon Artificial General Intelligence. The amazon nova family of models: Technical report and model card. *Amazon Technical Reports*, 2024. URL `https://www.amazon.science/publications/the-amazon-nova-family-of-models-technical-report-and-model-card`.

Yunseok Jang, Yale Song, Chris Dongjoo Kim, Youngjae Yu, Youngjin Kim, and Gunhee Kim. Video question answering with spatio-temporal reasoning. *International Journal of Computer Vision*, 127(10):1385–1412, 2019.

Baoxiong Jia, Ting Lei, Song-Chun Zhu, and Siyuan Huang. Egotaskqa: Understanding human tasks in egocentric videos. In *The 36th Conference on Neural Information Processing Systems (NeurIPS 2022) Track on Datasets and Benchmarks*, 2022.

Alexander Klaser, Marcin Marszałek, and Cordelia Schmid. A spatio-temporal descriptor based on 3d-gradients. In *BMVC 2008-19th British machine vision conference*, pp. 275–1. British Machine Vision Association, 2008.

Klaus Krippendorff. Computing krippendorff's alpha-reliability. 2011.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *Proceedings of the IEEE international conference on computer vision*, pp. 706–715, 2017.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pp. 611–626, 2023.

Ivan Laptev. On space-time interest points. *International journal of computer vision*, 64(2):107–123, 2005.

Jie Lei, Tamara L Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language queries. *Advances in Neural Information Processing Systems*, 34:11846–11858, 2021.

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.

Dongxu Li, Junnan Li, Hongdong Li, Juan Carlos Niebles, and Steven CH Hoi. Align and prompt: Video-and-language pre-training with entity prompts. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4953–4963, 2022.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023.

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, 2025.

Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. Hero: Hierarchical encoder for video+ language omni-representation pre-training. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pp. 2046–2065, 2020.

Yanwei Li, Chengyao Wang, and Jiaya Jia. Llama-vid: An image is worth 2 tokens in large language models. In *European Conference on Computer Vision*, pp. 323–340. Springer, 2024c.

Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3202–3211, 2022.

Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. Univl: A unified video and language pre-training model for multimodal understanding and generation. *arXiv preprint arXiv:2002.06353*, 2020.

Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval. *arXiv preprint arXiv:2104.08860*, 2021.

Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning. *Neurocomputing*, 508: 293–304, 2022.

Ruipu Luo, Ziwang Zhao, Min Yang, Zheming Yang, Minghui Qiu, Zhongyu Wei, Yanhao Wang, and Cen Chen. Valley: Video assistant with large language model enhanced ability. *ACM Transactions on Multimedia Computing, Communications and Applications*, 2023.

Yiwei Ma, Guohai Xu, Xiaoshuai Sun, Ming Yan, Ji Zhang, and Rongrong Ji. X-clip: End-to-end multi-grained contrastive learning for video-text retrieval. In *Proceedings of the 30th ACM international conference on multimedia*, pp. 638–647, 2022.

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. *NeurIPS*, 2023.

Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.

Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2630–2640, 2019.

Kepan Nan, Rui Xie, Penghao Zhou, Tiehan Fan, Zhenheng Yang, Zhijie Chen, Xiang Li, Jian Yang, and Ying Tai. Openvid-1m: A large-scale high-quality dataset for text-to-video generation. *arXiv preprint arXiv:2407.02371*, 2024.

OpenAI. GPT-4o mini: advancing cost-efficient intelligence. https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/, 2026a. Accessed: 2026-05-07.

OpenAI. Introducing GPT-5.4. https://openai.com/index/introducing-gpt-5-4/, 2026b. Accessed: 2026-05-07.

OpenAI. GPT-5.5 system card. https://openai.com/index/gpt-5-5-system-card/, 2026c. Accessed: 2026-05-07.

Viorica Patraucean, Lucas Smaira, Ankush Gupta, Adria Recasens, Larisa Markeeva, Dylan Banarse, Skanda Koppula, Joseph Heyward, Mateusz Malinowski, Yi Yang, Carl Doersch, Tatiana Matejovicova, Yury Sulsky, Antoine Miech, Alexandre Frechette, Hanna Klimczak, Raphael Koster, Junlin Zhang, Stephanie Winkler, Yusuf Aytar, Simon Osindero, Dima Damen, Andrew Zisserman, and Joao Carreira. Perception test: A diagnostic benchmark for multimodal video models. In *Advances in Neural Information Processing Systems*, 2023.

Hangyu Qin, Junbin Xiao, and Angela Yao. Question-answering dense video events. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '25, pp. 884–894. ACM, 2025. doi: 10.1145/3726302.3729945. URL http://dx.doi.org/10.1145/3726302.3729945.

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

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pp. 28492–28518. PMLR, 2023.

Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. Timechat: A time-sensitive multimodal large language model for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14313–14323, 2024.

Mattia Soldan, Alejandro Pardo, Juan León Alcázar, Fabian Caba, Chen Zhao, Silvio Giancola, and Bernard Ghanem. Mad: A scalable dataset for language grounding in videos from movie audio descriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5026–5035, 2022.

Tomás Soucek and Jakub Lokoc. Transnet v2: An effective deep network architecture for fast shot transition detection. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 11218–11221, 2024.

Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 7464–7473, 2019.

SYSTRAN. faster-whisper. `https://github.com/SYSTRAN/faster-whisper`, 2024.

StepFun Team. Step-3 is large yet affordable: Model-system co-design for cost-effective decoding, 2025. URL `https://arxiv.org/abs/2507.19427`.

The Gemini Team. Gemini 3.1 Pro: A smarter model for your most complex tasks. `https://blog.google/innovation-and-ai/models-and-research/gemini-models/gemini-3-1-pro/`, February 2026. Accessed: 2026-05-07.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Advances in neural information processing systems*, 35:10078–10093, 2022.

Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 4489–4497, 2015.

Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to sequence-video to text. In *Proceedings of the IEEE international conference on computer vision*, pp. 4534–4542, 2015.

Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *Proceedings of the IEEE international conference on computer vision*, pp. 3551–3558, 2013.

Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Dense trajectories and motion boundary descriptors for action recognition. *International journal of computer vision*, 103(1):60–79, 2013.

Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, pp. 20–36. Springer, 2016.

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.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025b.

Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7794–7803, 2018.

Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, et al. Internvideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*, 2022.

Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, et al. Internvid: A large-scale video-text dataset for multimodal understanding and generation. *arXiv preprint arXiv:2307.06942*, 2023.

Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14668–14678, 2022.

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.

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*, pp. 9777–9786, 2021.

LLM-Core-Team Xiaomi. Mimo: Unlocking the reasoning potential of language model – from pretraining to posttraining, 2025. URL https://arxiv.org/abs/2505.07608.

Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 6787–6800, 2021.

Antoine Yang, Arsha Nagrani, Ivan Laptev, Josef Sivic, and Cordelia Schmid. Vidchapters-7m: Video chapters at scale. *Advances in Neural Information Processing Systems*, 36:49428–49444, 2023.

Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B. Tenenbaum. CLEVRER: collision events for video representation and reasoning. In *ICLR*, 2020.

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.

Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. In *Proceedings of the 2023 conference on empirical methods in natural language processing: system demonstrations*, pp. 543–553, 2023.

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

Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 803–818, 2018a.

Luowei Zhou, Chenliang Xu, and Jason Corso. Towards automatic learning of procedures from web instructional videos. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018b.

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.

Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8746–8755, 2020.

APPENDIX

A RELATED WORK

Video Understanding and Video-Language Models. Early video understanding relied on hand-crafted spatiotemporal descriptors and trajectory-based representations (Laptev, 2005; Klaser et al., 2008; Wang & Schmid, 2013; Wang et al., 2013). These methods established the fundamentally spatiotemporal nature of video analysis, but they typically produced clip-level decisions rather than persistent representations of entities and states over time. Deep architectures later shifted the field toward learnable spatiotemporal features (Tran et al., 2015; Carreira & Zisserman, 2017; Feichtenhofer et al., 2019; Wang et al., 2016; Zhou et al., 2018a; Wang et al., 2018). While these models substantially improved video representations, their supervision was still largely aligned with clips or

full videos rather than explicitly with second-level state transitions or temporally consistent entity attributes. Video Transformers further improved long-range temporal modeling through native cross-frame attention (Bertasius et al., 2021; Arnab et al., 2021; Liu et al., 2022), while self-supervised pretraining methods demonstrated strong transfer without dense annotation (Tong et al., 2022; Wei et al., 2022; Akbari et al., 2021; Wang et al., 2022). Nevertheless, much of this literature still optimizes coarse temporal objectives and often relies on sparse temporal sampling or compressed visual tokens for efficiency, which can weaken sensitivity to subtle state changes in long videos.

Video-language research evolved from sequence-to-sequence captioning models such as S2VT (Venugopalan et al., 2015) to large-scale video-text pretraining frameworks (Sun et al., 2019; Zhu & Yang, 2020; Li et al., 2020; Luo et al., 2020; Fu et al., 2021; Li et al., 2022; Xu et al., 2021). Large web-scale corpora such as HowTo100M (Miech et al., 2019) and WebVid (Bain et al., 2021), together with CLIP-style contrastive learning (Radford et al., 2021), enabled more scalable cross-modal alignment and inspired retrieval-oriented extensions (Luo et al., 2021; 2022; Ma et al., 2022; Fang et al., 2021; Bain et al., 2021). However, these corpora are often only weakly aligned in time, and most training objectives emphasize global or clip-level matching rather than second-level, temporally anchored scene grounding. Recent Video-LLMs connect visual encoders to large language models through adapters, query modules, or projection layers. Connector-based architectures such as Flamingo (Alayrac et al., 2022) and BLIP-2 (Li et al., 2023) established scalable multimodal interfaces, and subsequent systems, including Video-LLaMA (Zhang et al., 2023), Video-ChatGPT (Maaz et al., 2024), LLaVA-Video (Li et al., 2024a), LLaMA-VID (Li et al., 2024c), and TimeChat (Ren et al., 2024), extended this paradigm to video dialogue, instruction following, and long-context reasoning. Even so, a central tension remains between temporal coverage and spatiotemporal fidelity: sparse frame sampling, frame pooling, and aggressive token reduction improve throughput, but they can hinder fine-grained temporal localization, entity tracking, and temporally consistent reasoning.

Video-Language Datasets and Benchmarks. Large-scale video-text pretraining datasets provide the foundation for many modern video-language models. Web-scale corpora such as WebVid (Bain et al., 2021), InternVid (Wang et al., 2023), and OpenVid-1M (Nan et al., 2024) collect millions to hundreds of millions of video-text pairs, enabling scalable video-text representation learning. These datasets improve coverage and diversity, but their annotations are weakly aligned and lack dense temporal supervision. More recent high-quality captioning datasets, such as ShareGPT4Video (Chen et al., 2024b), and FineVideo (Farré et al., 2024), move toward richer and more structured supervision. Another important line of work focuses on video instruction tuning and conversational video data. Datasets such as VideoChat-11K (Li et al., 2025), Video-ChatGPT-100K (Maaz et al., 2024), Valley-Instruct-65K (Luo et al., 2023), LLaVA-Video-178K (Zhang et al., 2024), ShareGPTVideo (Chen et al., 2024b), and TimeIT (Ren et al., 2024) transform existing captions, QA annotations, or video understanding tasks into instruction-following formats. While these datasets provide richer language supervision, much of their annotations are still coarse in spatial and temporal granularity. As a result, they lack fine-grained grounding in space and time and provide limited supervision over video dynamics.

Temporal grounding and dense video understanding benchmarks provide more explicit temporal supervision. Classic and widely used datasets such as ActivityNet (Krishna et al., 2017), Charades-STA (Gao et al., 2017), DiDeMo (Anne Hendricks et al., 2017), and QVHighlights (Lei et al., 2021) annotate the relationship between natural language queries and temporal segments in videos. These benchmarks have played an important role in moving video-language evaluation beyond full-video classification and toward timestamp-aware grounding. More recent long-form or egocentric grounding resources, such as MAD (Soldan et al., 2022) and Ego4D (Grauman et al., 2022), extend this setting to movies or first-person videos. VidChapters-7M (Yang et al., 2023) further introduces large-scale chapter-level supervision for long videos, with chapter titles and timestamps collected from web videos. These datasets are highly relevant to temporal localization, but they still do not fully solve the problem of fine-grained state and attribute tracking. Moment boundaries are often coarse, query-level, or segment-level, and the annotations usually describe events or steps rather than maintain a persistent inventory of entities, attributes, relations, and state changes across time.

Video question-answering benchmarks have also evolved from short-clip QA toward long-form and diagnostic evaluation. Earlier benchmarks evaluate video understanding through manually annotated or carefully constructed questions, including both open-ended and multiple-choice formats (Yu et al.,

2019; Xiao et al., 2021; Grauman et al., 2022). More recent comprehensive benchmarks, including MVBench (Li et al., 2024b), LongVideoBench (Wu et al., 2024), Video-MME (Fu et al., 2025), and LVBench (Wang et al., 2025a), expand evaluation toward multi-task reasoning, long-video understanding, and full-spectrum video modeling. These benchmarks have been crucial for revealing the limitations of Video LLMs under long-context settings, but most of them still reduce evaluation to discrete multiple-choice accuracy. This makes leaderboards easy to compare, but it can also obscure whether a model truly tracks temporal evidence, maintains entity consistency, or merely exploits language priors and coarse scene summaries.

Several diagnostic and multi-task benchmarks aim to evaluate more detailed perceptual and temporal capabilities. The Perception Test includes object tracks, point tracks, action segments, sound segments, multiple-choice video QA, and grounded video QA, providing a more diverse testbed for perception, grounding, and multimodal reasoning (Patraucean et al., 2023). Datasets such as STAR, CLEVRER, EgoTaskQA, and TGIF-QA focus on situated reasoning, physical reasoning, egocentric reasoning, state transitions, and repetition counting (Wu et al., 2021; Yi et al., 2020; Jia et al., 2022; Jang et al., 2019). These datasets isolate specific reasoning factors but are typically built on short or synthetic clips and thus complement rather than replace long-video benchmarks that require sustained temporal reasoning.

Overall, existing datasets and benchmarks have substantially advanced video-language learning, but they leave an important gap for fine-grained, temporally faithful video understanding. Large-scale web corpora provide breadth but only weak temporal and spatial alignment. Instruction datasets improve the conversational interface of VideoLLMs but often rely on synthetic or model-generated supervision. Temporal grounding datasets provide timestamp supervision but usually focus on query-to-moment localization rather than persistent entity and state modeling. Long-video QA benchmarks expose the difficulty of reasoning over extended contexts, but their multiple-choice format can underdiagnose perceptual failures, temporal hallucinations, and entity-state inconsistencies. These limitations suggest the need for benchmarks and training resources that combine long temporal coverage with dense temporal anchoring, explicit spatial and entity-level grounding, and evaluation protocols that test not only whether a model gives the correct answer, but also whether it can maintain temporally consistent, evidence-grounded representations throughout the video.

B Annotation Pipeline Hyperparameters

This section enumerates the hyperparameters used by every pipeline stage so that the corpus and the benchmark can be reproduced exactly. The four artifacts emitted per video are described in Table 5 of §C; the present section concerns *how* those artifacts are produced.

B.1 Shot Detection and Frame Sampling

Shot detector. Shot boundaries are produced by TransNetV2 (Soucek & Lokoc, 2024) with a softmax threshold of 0.5 and a minimum scene length of 15 source frames. The threshold is intentionally slightly recall-biased: a missed boundary merges two semantically distinct shots and breaks every cross-shot question generated for the resulting merged span, whereas a spurious boundary at most introduces a redundant reference frame inside a real shot. Both hard cuts and gradual transitions are accepted as boundaries; the gradual-transition probability head is read from the same TransNetV2 forward pass.

Frame sampling. After shot detection, the pipeline materialises a per-frame structural record and extracts a 1 FPS JPEG subset for downstream annotation. Frames are resized so that the longer image dimension is 1024 pixels, and saved at JPEG quality 90 . The resulting 1 FPS timeline is the spine of the annotation stream and is what every subsequent description, audio segment, and benchmark evidence span is keyed to.

Reference and differential frames. Within each shot, the first extracted frame is treated as the *initial frame* and receives a full spatial description plus, when entity tracking is enabled, a structured entity list. All later 1 FPS samples in the same shot are treated as *differential frames* and are described only in terms of what changed relative to the previous state. For long shots, the pipeline re-anchors every 300 differential frames by promoting the current frame to a new initial frame, which keeps prompt context bounded and reduces drift in long description chains.

B.2 AUDIO PIPELINE

Two complementary audio models. Audio is processed by two complementary models running on the same physical device. Speech is transcribed with FASTER-WHISPER-LARGE-V3 (SYSTRAN, 2024; Radford et al., 2023) at sentence resolution. Non-speech audio is summarized by QWEN2-AUDIO-7B-INSTRUCT (Chu et al., 2024) over fixed 30 -second windows; this captures environmental and event-level sounds such as crowd noise, engine sounds, applause, footsteps, and music.

Alignment. The two audio streams are aligned to the same 1 FPS timeline used by the visual annotation stage. Each speech sentence is attached to the first processed frame whose timestamp falls inside the sentence's time span, and never attached twice. Environmental audio summaries are attached only to initial frames, not to every differential frame, so that nearby records are not burdened with repeated ambient descriptions.

B.3 VISUAL ANNOTATION PROMPTS

Server. The visual annotation stage is driven by QWEN3-VL-8B-INSTRUCT served through vLLM (Kwon et al., 2023) with tensor parallel size 2 , `bfloat16` weights, sampling temperature 0.2 , and `gpu_memory_utilization` 0.85 .

Four prompt templates. The pipeline emits descriptions through four prompt templates:

- `NARRATIVE_REF_WITH_ENTITIES` (initial frames). Returns structured JSON of the form {`"description": ..., "entities": [...]`}: a dense paragraph plus an entity list with canonical mention, entity type, and a `visual_details` string for re-identification. Conditioned on the current frame, the current entity bank, and a narrative context window of the previous 20 shot descriptions.
- `DIFFERENTIAL` (within-shot differential frames). Receives the previous and current frame plus a shot-local description chain capped at the previous 50 descriptions. Forbids restating already-described static content; allows an empty string when no meaningful change is observed.
- `TRANSITION` (shot boundaries). Receives the two boundary frames of adjacent shots together with their shot descriptions, and asks the model to describe the cut in terms of editing technique, narrative purpose, and framing change without restating scene content already covered.
- `REFERENCE` (fallback). A plain single-frame captioning path used when entity-structured output is not needed.

Audio injection. Aligned ASR and environmental audio are appended to every prompt as a shared suffix that contains only the not-yet-used segments anchored to the current timestamp; this brings spoken and ambient content into the same frame-level annotation stream without a separate fusion stage.

B.4 CROSS-SHOT ENTITY MATCHING

Entity vocabulary. For each initial frame the model emits a structured set of entities drawn from the fixed vocabulary of 8 categories: `person`, `animal`, `object`, `vehicle`, `text`, `location`, `food`, `clothing`. Every instance carries a canonical phrase and a concise `visual_details` string intended to preserve identifying appearance cues across shots.

Label-exact matching. Cross-shot entity assignment relies on the prompt contract rather than on a similarity threshold. The current bank is rendered into each initial-frame prompt as one record per entity (canonical name, type, and `visual_details`), limited to the 400 most recently seen entities, and the model is instructed to reuse a bank entity's canonical name verbatim. An emitted mention is matched against the bank by normalized label equality: lower-casing, removal of a leading article and of edge punctuation, and whitespace collapsing. A hit appends an appearance (shot, frame, surface phrase) to the existing entity; a miss registers a new entity ID.

B.5 MEGA-BATCH INFERENCE

Single batch across the workload. Each inference round collects the initial-frame prompts of at most one shot per video (a long shot re-anchored every 300 frames contributes one prompt per

Table 5: **Per-video output schema.** `descriptions.jsonl` is the central supervision artifact; the other five files carry structural metadata that supports downstream sampling.

File	Producer	Role
`frames.tar`	`prep`	1 FPS JPEG samples ($\leq$1024 px, JPEG quality 90), packed into one tar per video.
`prep_cache.json`	`prep`	shot list and frame manifest, fingerprinted by the shot-detector and sampler config.
`audio_segments.json`	`audio`	FASTER-WHISPER-LARGE-V3 speech segments and QWEN2-AUDIO-7B-INSTRUCT environment summaries.
`descriptions.jsonl`	`infer`	per-frame reference and differential descriptions; `transition` field on shot boundaries.
`entities_final.json`	`infer`	per-video entity bank: canonical label, type, visual details, first appearance.
`resume_state.json`	`infer`	entity counter and ASR deduplication state; checkpointed at every shot boundary.

segment) and then fills the remaining token budget with differential prompts drawn from segments whose initial frame has already been completed. Prompt lengths are estimated analytically before batching and any prompt that would overflow the round budget is deferred. The resulting workload is submitted through one `vllm.generate_batch` call.

Token budget and engine cadence. The hard cap is 1.2M prompt tokens per round, large enough to saturate both inference GPUs while remaining stable against out-of-memory spikes from heavy multi-image prompts. The vLLM engine is rebuilt every 150 rounds so that shared-memory artifacts that accumulate over long tensor-parallel sessions do not destabilise multi-hour annotation jobs.

B.6 HARDWARE AND THROUGHPUT

The pipeline runs on a homogeneous H200 cluster. The `prep` and `audio` stages each exceed $20\times$ real time on a single H200, while the `infer` stage reaches roughly $4\times$ real time per H200, or $8\times$ aggregate throughput under tensor-parallel-2 serving.

C PER-VIDEO SCHEMA AND DISTRIBUTION DIAGNOSTICS

C.1 PER-VIDEO SCHEMA

The pipeline writes the six artifacts listed in Table 5 for every video; four of them are released, while `frames.tar` is regenerated from the source video and `resume_state.json` is an internal checkpoint. The central supervision channel is `descriptions.jsonl`: one JSON record per 1 FPS extracted frame, containing the absolute timestamp, the shot identifier and shot start/end frames and times, a flag `is_reference` distinguishing reference frames from differential frames, the description text, and, on reference frames that sit at a shot boundary, a populated `transition` field carrying the cross-shot dynamics description. This single record type is sufficient to support every axis of the derived benchmark and every granularity of the fine-tuning corpus — a downstream sampler need only filter by `is_reference` and the presence of `transition` to obtain the artifact it wants.

C.2 SOURCE CURATION

Two-platform pool, three-level taxonomy. Videos are drawn from two public long-form video platforms, YouTube and Bilibili, seeded by two curated download lists that together pool 28,282 candidate URLs (15,054 YouTube + 13,228 Bilibili). Each row is pre-tagged with a three-level content taxonomy: 12 parent *domains* (`A. Sports`, `B. Gaming`, `C. Ego-Centric & Daily Life`, `D. Vlogs & Ceremonies`, `E. Arts & Crafts`, `F. Media & Entertainment`, `G. Public Safety`, `H. Embodied AI`, `I. Drones & Remote Sensing`, `J. AIGC-related Content`, `K. Formal Communication`, `L. Computer Use`); 35 *categories* nested within domains; and 202 *scenarios* at the leaf level (e.g.

Basketball within `A. Sports / I. Ball Games`). The full taxonomy — every leaf scenario under its parent (domain, category) — is enumerated in Table 6.

Table 6: **Full 12 -domain / 35 -category / 202 -scenario content taxonomy of KAIROS.** Each candidate URL in the 28,282 -URL pool was pre-tagged with one (domain, category, scenario) triple at curation time. The same taxonomy carries through to the annotation corpus and to the benchmark sampler, so any per-axis evaluation cut can be re-grouped by content type.

Domain	Category	Scenarios
A. Sports	I. Ball Games	Basketball, Soccer/Football, Volleyball, Baseball, American Football, Golf, Snooker, Bowling
	II. Racket Sports	Tennis, Badminton, Table Tennis
	III. Water & Ice Sports	Swimming, Diving, Water Polo, Ice Hockey, Curling, Figure Skating, Speed Skating, Short Track
	IV. Athletics	Racing, Relay Race, Marathon, Long Jump, High Jump, Hurdles, Pole Vault, Shot Put, Discus Throw, Javelin Throw, Hammer Throw
	V. Gymnastics	Artistic Gymnastics, Trampoline
	VI. Combat Sports	Boxing, Wrestling, Judo, Taekwondo, Karate, MMA, WWE, Kickboxing, Wushu, Fencing, Sumo
	VII. Racing Sports & Equestrian	Formula 1, Rally Racing, Off-road Racing, Motorcycle Racing, Horse Racing, Equestrian
	VIII. Extreme Sports	Skateboarding, BMX, Parkour, Surfing, Alpine Skiing, Snowboarding, Rock Climbing, Bouldering, Skydiving, Bungee Jumping, Scuba Diving
B. Gaming	I. Video Games	First-Person Shooter, MOBA, VR Games, Soulslike, Roguelike, Sandbox Games, Racing Simulation, Sports Simulation, Horror Games, Puzzle Games
	II. Board & Strategy Games	Go, Chess, Chinese Chess, Gomoku, Mahjong, Poker, Bridge
C. Ego-Centric & Daily Life	I. Household Activities	Cooking, Washing Dishes, Folding Laundry, Ironing Clothes, Vacuuming, Assembling Furniture, Fixing Furniture, Cleaning, Child Care
	II. Daily Skills	Typing, Handwriting, Tool Using, Equipment Operation, Playing Instruments, Conversation, Studying, Teaching, First Aid, Car Repair, Tire Change
	III. Personal Care	Hand Washing, Makeup, Skincare, Taking Medicine, Rehabilitation
	IV. Outdoor Activities	Gardening, Agricultural Labor, Tourism
D. Vlogs & Ceremonies	I. Vlogs	Daily Vlogs, Travel Vlogs, Walking Vlogs, Running Vlogs, Riding Vlogs, Shopping Vlogs, Gym Vlogs
	II. Ceremonies	Wedding Ceremony, Birthday Party
E. Arts & Crafts	I. Visual Arts	Pencil Sketching, Oil Painting, Watercolor Painting, Calligraphy
	II. Handcraft	Pottery, Origami, Paper Cutting, Embroidery, Knitting, Woodworking, Sculpting, Jewelry Making, Glass Blowing, Leather Crafting, 3D Printing
F. Media & Entertainment	I. Drama Genres	Medical, Legal, Crime, Domestic, School, Xianxia, Spy, Office
	II. Shows & Performance	Talk Show, Stand-up Comedy, Magic, Street Performance, Circus, Acrobatics, Concert
	III. News & Documentary	News Broadcast, Documentary
	IV. Musical & Opera	Musical, Opera
	V. Animation	Animated Films, Animated Series
G. Public Safety	I. Driving	Urban Driving, Highway Driving
	II. Surveillance	Traffic Surveillance, Public Space Monitoring, Home Security

(continued on next page...)

(...continued from previous page)

Domain	Category	Scenarios
H. Embodied AI	I. Embodied Interaction	Manipulation, Tool Use, Navigation, Human-Robot Interaction, Dexterous Manipulation
I. Drones & Remote Sensing	I. Aerial Video	Aerial Footage, Survey Flights
	II. Satellite Video	Satellite Video, Time-lapse Observation
J. AIGC-related Content	I. Generated Content	Text-to-Video Samples, AI Stylized Animation, Deepfake
	II. Artifacts & Consistency	Compression Artifacts, Moiré, AI Motion Glitch, Frame Interpolation Artifacts, Temporal Consistency Issues
K. Formal Communication	I. Academic	Conference Presentation, Plenary Speech, Poster Presentation, Job Talk, Seminar, Group Meeting, Dissertation Defense, TED-style Talk, Lecture
	II. Business	Investor Pitch, Product Launch, Board Meeting, Contract Negotiation
	III. Politics	Stump Speech, Acceptance Speech, State of the Union, Press Conference, Legislative Debate, Diplomatic Negotiation, Parliamentary Session
L. Computer Use	I. Software Tutorial	Word, Excel, PowerPoint, PhotoShop, LightRoom, Premiere, Blender, VS Code
	II. Daily Activities	Browse, E-mail

Duration filtering. Candidate URLs are filtered to the 10 –30 minutes duration band: the length regime where clip-scale benchmarks stop scaling and where the Time axis becomes non-trivial. Sources longer than 30 min are split into consecutive fixed-length 30-minute parts (the `_partNN` suffixes visible in the corpus) rather than truncated, so no footage is discarded. After duration filtering and download-side failures (privacy takedowns, geo-restrictions, channel deletion), the final annotation corpus contains 19,004 videos totalling 5,420 hours.

C.3 Length and Structure

The corpus's content taxonomy and length distribution are summarized in Figs. 6–9, with the deepest per-scenario decomposition in Fig. 10. Together these substantiate the design claim that KAIROS lives in the *multi-shot, long-context* regime that single-clip benchmarks do not exercise.

D Benchmark Construction Details

D.1 Source-type Samplers and Tier Mapping

Each of the 10 source types corresponds to a single sampler over the timestamped annotation stream. A sampler returns the materials needed for question generation — the relevant annotations, their timestamps, the evidence span, neighbouring reference-frame context, and a pool of candidate distractors — without ever accessing the raw frames. The temporal tier of a generated question is determined by the realized evidence span itself, not assigned post hoc by the LLM. Each sampler also restricts the capability labels that the generator may emit (e.g. `ref_audio` can only be tagged with `A5_audio_comprehension`, while `transition` is tagged with `C2_narrative_transition`). Table 7 reports the per-source-type composition of the released benchmark, alongside the tier range that each sampler's evidence span can fall into.

Atomic generation. Given the sampled evidence, the neighbouring reference-frame context before and after the target evidence, and a coarse temporal hint, GEMINI-2.5-FLASH returns a structured JSON object containing four fields: `question`, `answer`, `distractors`, and `reasoning`. All four are produced atomically in the same call. This encourages internal consistency between the answer and the reasoning, and avoids a separate rewriting stage that could make the distractors superficially different from the correct answer.

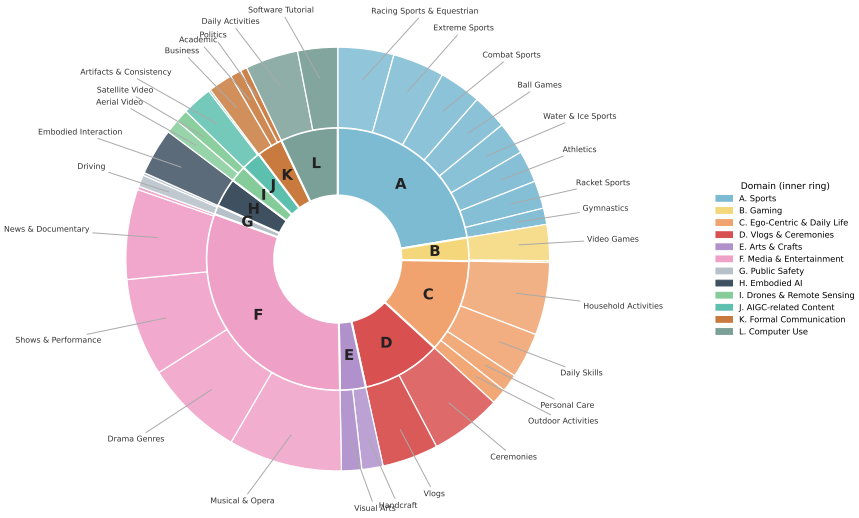

Figure 6: **Two-ring content taxonomy of the KAIROS corpus.** Inner ring: the 12 parent domains (codes A–L). Outer ring: the 35 categories nested within domains, with leader-line labels. Wedge sizes are video counts. The corpus is non-uniform but well-spread: `A. Sports` and `F. Media & Entertainment` together account for the largest share, while small-tail domains (`G. Public Safety`, `H. Embodied AI`, `I. Drones & Remote Sensing`) are kept at single-digit shares so that downstream evaluation can isolate domain-specific failures.

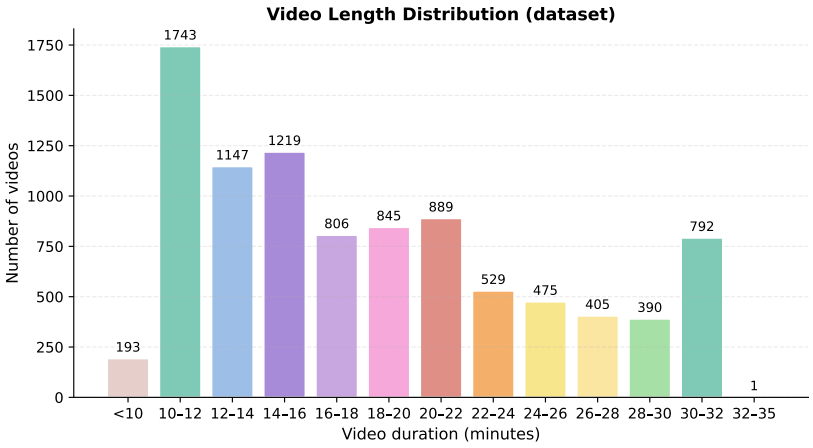

Figure 7: **Per-video duration distribution** on the YouTube partition (9,434 videos, 2,857 hours). Median 17.0 min, mean 18.2 min, max 32.7 min. The bulk of the corpus sits in the 10 –30-min target band; the small `<10` bar reflects `_partNN` tail fragments left after splitting longer sources rather than truncation.

D.2 CROSS-BENCHMARK TEXT-ONLY LEAKAGE — PROTOCOL

The cross-benchmark leakage probe in Table 1 of the main text fixes the solver, the prompt, the option-shuffle protocol, and the sample size identically across all benchmarks:

- **Solver.** GEMINI-3.1-PRO , no video and no audio, $K{=}1$ shuffle, seed 42, greedy decoding, with the smallest thinking budget the model accepts (128 tokens).
- **Sample size.** 500 questions per benchmark, stratified over capability/tier/source where applicable, otherwise uniform.

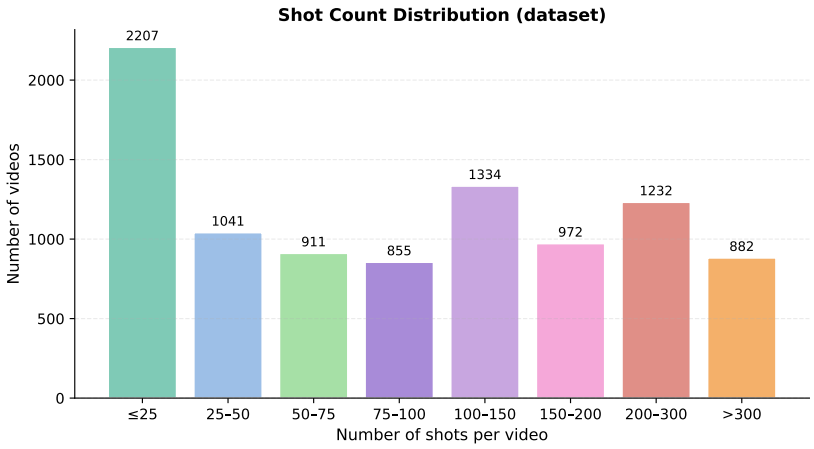

Figure 8: **Shots-per-video distribution** on the YouTube partition. Median 91 shots per video, mean 126.4 , max 1,235 , and 1,192,552 shots in total. The right tail is dominated by fast-cut sports broadcasts and clip compilations and motivates the differential-frame chain in the annotation pipeline.

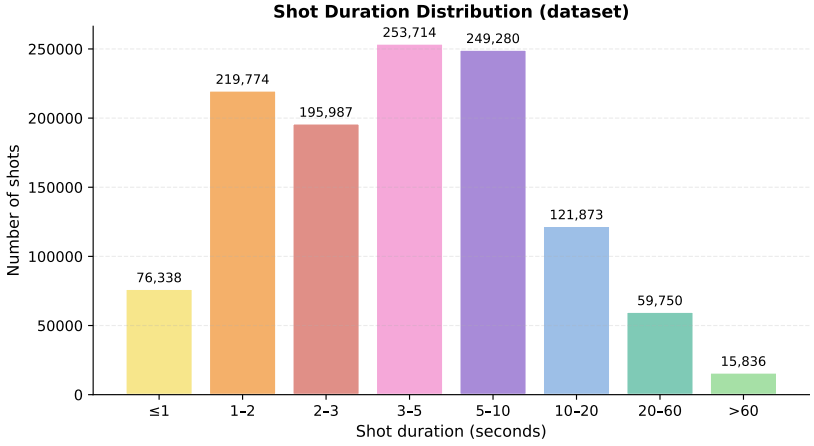

Figure 9: **Per-shot duration distribution** on the YouTube partition. Median 3.7 s, mean 8.6 s. The right tail (held shots, >60 s) is what makes Time-axis questions non-trivial in the derived benchmark, and is also where the 300 -frame re-anchor schedule is exercised.

- **Prompt.** A fixed system prompt states that no video or images are available and asks for the answer letter on the first line followed by one sentence of rationale on the second; the user message contains the question stem and the shuffled options. No in-context examples and no reasoning before the answer.
- **Random baseline.** Computed per benchmark from the realized option counts ($1/N$ for fixed-N benchmarks, and the per-question mean of $1/N_i$ for variable-option benchmarks).
- **Leakage.** Solver accuracy minus random baseline, reported in percentage points.

Figure 11 fits a linear regression of leakage on average total characters across the ten benchmarks; KAIROS-Bench has the longest stems among the four-option benchmarks but lies *below* the regression line, i.e. its leakage is lower than its length would predict. Figure 12 shows the underlying scatter, broken down by benchmark.

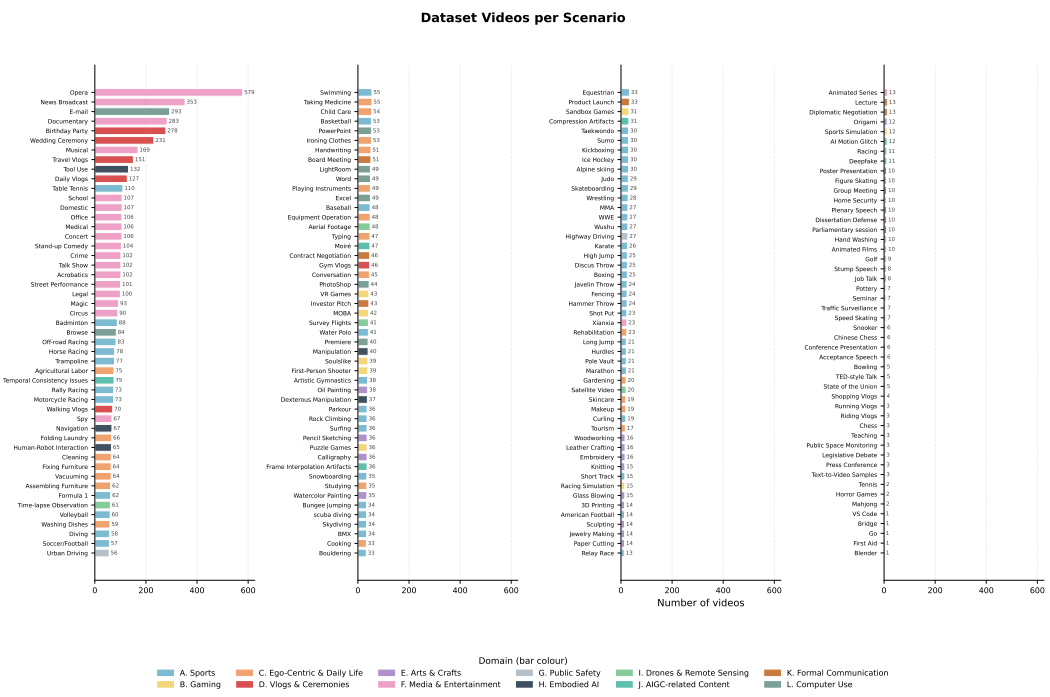

Figure 10: **Per-scenario video count** across all 202 scenarios of the corpus, grouped by parent category and parent domain (bar color). Bars are sorted within each category by descending count; the resulting distribution is long-tailed but non-degenerate, with no single scenario dominating and every scenario receiving a meaningful share of the corpus.

Table 7: **Per-source-type composition of KAIROS-Bench** after audit and human review. *Allowed tiers* is the range of evidence spans the sampler can realise; the tier of a question is set by the realized span, not by the source type.

Source type	Evidence shape	Allowed tiers	#Q
`ref_perception`	single reference frame, scene + entities + relations	T1	614
`ref_ocr`	single reference frame, on-screen text	T1	177
`ref_audio`	single reference frame, aligned ASR / environment	T1	184
`diff_change`	ref + several differential frames inside one shot	T2	457
`diff_sequence`	full differential chain of one shot	T2	555
`entity_tracking`	cross-shot entity reappearance	T2–T5	366
`transition`	two boundary frames + adjacent shot descriptions	T1–T4	53
`cross_shot`	several adjacent or near-adjacent shots	T2–T5	236
`long_range`	evidence span >180 s	T3–T4	144
`full_video`	whole-video integration	T4–T5	225

D.3 BENCHMARK DISTRIBUTION DIAGNOSTICS

Figures 13–18 expand the three-axis summary in Fig. 4 of the main text to the content and structural axes, none of which are exposed in the main paper. Figure 13 reframes the content taxonomy as a two-ring pie (domain $\rightarrow$ category) so that the relative weight of each domain is legible. Figure 14 reports the benchmark question count by content category, and Fig. 15 the per-scenario decomposition; both confirm that the 3,011 benchmark questions preserve the corpus-level taxonomy distribution rather than over-sampling a small subset. Figures 16–18 report per-video duration, shots per video, and per-shot duration on the benchmark subset.

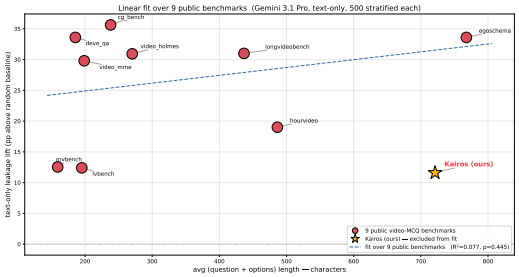

Figure 11: **Leakage vs. average MCQ length.** Linear fit of text-only-leakage (pp) on average total characters per question across the ten benchmarks of Table 1 of the main text. KAIROS-Bench is to the right of the cloud (long stems) and below the line (lower leakage than length predicts).

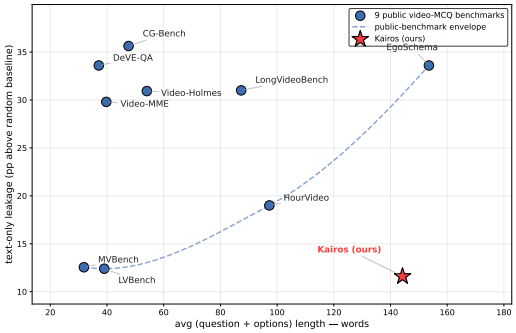

Figure 12: **Leakage–length scatter.** Per-benchmark scatter underlying the fit in Fig. 11. Each dot is one of the ten benchmarks; axes are average total MCQ characters and the absolute lift over the random baseline. KAIROS is the rightmost low-leakage point.

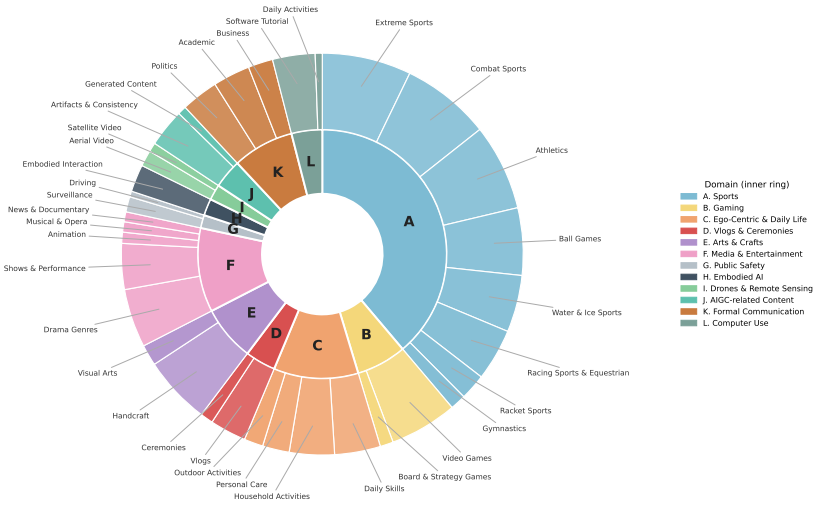

Figure 13: **Two-ring content taxonomy of KAIROS-Bench.** Inner ring: the 12 parent domains (codes A–L). Outer ring: the 35 categories nested within domains, with leader-line labels. Wedge sizes are question counts.

E EVALUATION PROTOCOL

We evaluate 21 models in total: 7 closed-source proprietary and 14 open-weight. A unified runner groups questions by `video_id`, prepares the video once per video for the relevant backend, then dispatches every question for that video in parallel (API: ThreadPoolExecutor) or batched (vLLM: `generate_batch`).

E.1 BACKENDS AND VIDEO-INPUT NEGOTIATION

Five backends. The runner instantiates one of five backends per model based on its registry entry:

- **vLLM** (open-weight): frame mode for every family except LLAVA-VIDEO-7B, which is fed the video directly.
- **HuggingFace Transformers** (open-weight): used for the few open-weight models without a vLLM video implementation, in frame mode only.

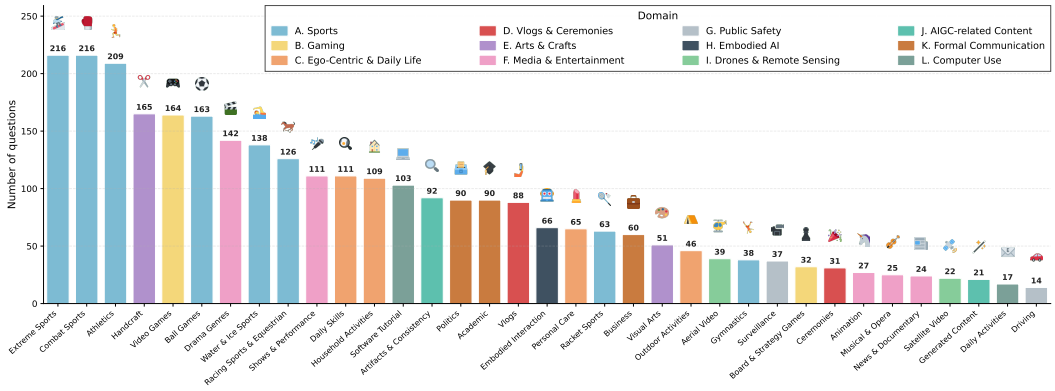

Figure 14: **Per-category question count of KAIROS-Bench** (3,011 questions), grouped by parent domain (bar color). The benchmark preserves the long-tail shape of the corpus (Fig. 3 of the main text); no domain is dropped, and the small-tail domains `G`, `H`, `I` are over-sampled relative to their share of the corpus to keep per-domain question counts non-trivial.

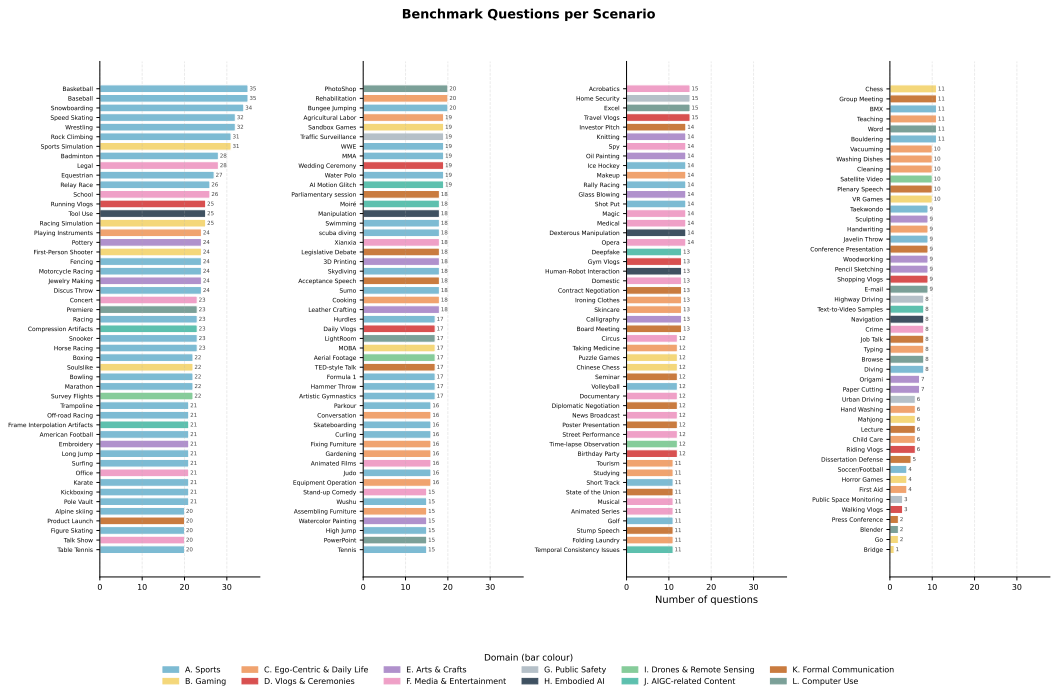

Figure 15: **Per-scenario question count of KAIROS-Bench** across the 202 scenarios of the content taxonomy. Bars are colored by parent domain. The benchmark question pool is intentionally spread thin across scenarios rather than concentrated in a few high-volume cells, so that per-scenario evaluation is well-defined for as many cells as possible at the cost of small absolute counts in the long tail.

- **OpenAI direct** (closed-source + OpenRouter-proxied open-weight): probes `data:video/mp4` native video once per model and caches the result; falls back to N uniformly subsampled frames otherwise.
- **Gemini direct**: File API upload + 2 s processing poll, cached per video path. Native video is the only input mode.
- **Anthropic direct**: frame mode only (no native video support in the public API).

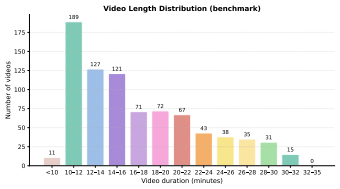

Figure 16: **Per-video duration** on the 820 benchmark videos.

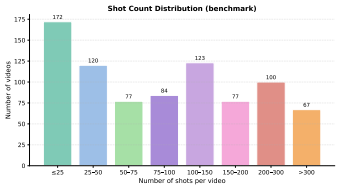

Figure 17: **Shots per benchmark video.**

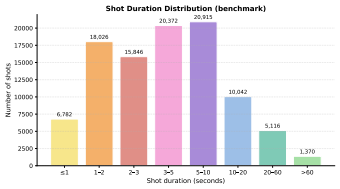

Figure 18: **Per-shot duration** on the benchmark videos.

Table 8: **Per-model evaluation configuration.** *Type* is the backend used. *TP* is the tensor-parallel size on H200 GPUs (vLLM only). *Frames* is the per-question frame budget (*native* = native video input). *Max tokens* is the decoding cap. Sampling is greedy throughout.

Model	Type	TP	Frames	Max tokens	Notes
Closed-source proprietary					
GEMINI-3.1-PRO	gemini	—	native	1024	video-native
GEMINI-2.5-FLASH	gemini	—	native	512	video-native
GPT-5.5	openai	—	256	2048	forced frame mode
GPT-4O	openai	—	64	1024	forced frame mode
GPT-4O-MINI	openai	—	128	1024	forced frame mode
NOVA-2-LITE	openai	—	32	1024	forced frame mode
SEED-2.0-LITE	openai	—	32	1024	forced frame mode
Open-weight					
INTERNVL3-78B	vLLM	4	16	512	frame mode
INTERNVL3.5-38B	vLLM	2	12	512	frame mode
INTERNVL3-8B	vLLM	2	16	512	frame mode
INTERNVL3.5-8B	vLLM	2	12	2048	structured-output letter
GLM-4.5V	vLLM	4	16	64	`enable_thinking=False`
GLM-4V-9B	HF	—	1	512	1-frame ceiling per the model card
STEP3-VL-10B	vLLM	2	16	8192	frame mode
QWEN3-VL-30B-A3B	vLLM	2	16	512	frame mode
QWEN3-VL-8B	vLLM	1	16	512	frame mode
QWEN2.5-VL-7B	vLLM	1	16	512	frame mode
MIMO-VL-7B	vLLM	1	16	4096	structured-output letter
COGVLM2-VIDEO-13B	HF	—	16	1024	frame mode
GEMMA-4-31B	openai	—	32	1024	forced frame mode
LLAVA-VIDEO-7B	vLLM	1	native	512	video-native; structured-output letter

Native-video models. Only GEMINI-2.5-FLASH, GEMINI-3.1-PRO, LLAVA-VIDEO-7B run with native video input in our evaluation; every other model receives uniformly subsampled frames at the per-model budget reported in Table 8.

E.2 PER-MODEL DECODING AND FRAME BUDGET

Table 8 enumerates the eval-time configuration used for every model in the leaderboard of Table 2 of the main text. The frame budget is the maximum number of frames the runner sends per question; the actual count for a given question equals $\min(\textit{budget}, \lfloor \textit{video_dur} \cdot 1 \rfloor)$ so that very short videos are never up-sampled past their native 1 FPS rate. Decoding is greedy (temperature 0) for every model except GPT-5.5, whose API only accepts its default sampling temperature; for the OpenQA run we keep the same decoding so that the only intentional change between MCQ and OpenQA is the answer format.

Scoring. A model output is parsed by first stripping any `<think>` block, then taking the last explicit `Answer:X` if present, otherwise a leading bracketed or bare letter; outputs with no such letter are counted as incorrect. Per-tier, per-capability, per-source, and per-domain accuracies are computed by simple bucket-then-average over the parsed letters. Three open-weight models (INTERNVL3.5-8B, MIMO-VL-7B, LLAVA-VIDEO-7B) are decoded with vLLM

Table 9: **Fine-tuning hyperparameters.** The recipe uses LLaMA-Factory's `qwen2_5_vl` template; the seven LoRA target modules are the standard transformer projections.

Setting	Value
Model and parameterisation	
Base model	QWEN2.5-VL-7B-INSTRUCT
Framework	LLaMA-Factory
Adapter	LoRA, rank 32 , α =64
LoRA target modules	`q,k,v,o,gate,up,down_proj`
Frozen modules	vision tower (multi-modal projector trainable)
Precision	bf16
Distributed	DeepSpeed ZeRO-3
Hardware	4×H200
Optimization	
Optimizer	AdamW (LLaMA-Factory default)
Learning rate	1×10^{-4}
Schedule	cosine, warmup ratio 0.03
Weight decay	0.01
Per-device batch size	1
Gradient accumulation	8
Effective batch size	32
Epochs	1
Cutoff length	16,384 tokens
Visual input	
Frame budget per sample	32
Sampler	uniform
Input resolution	420×420 pixels per frame (image and video pixels)
Frame source	pre-extracted 1 FPS JPEGs (no re-decoding)

`structured_outputs` that constrain the final token to a single letter, because their unconstrained outputs occasionally trail off into reasoning without committing to a choice.

E.3 OPEN-ENDED QA JUDGE

Why an OpenQA pass. A multiple-choice task is easy to score but lossy: a model can recognise the right answer without being able to produce it. We therefore additionally run an OpenQA pass on the same 3,011 questions, asking each model to generate a free-form answer rather than pick a letter. Decoding stays greedy; the per-model token budget is raised to at least 256 tokens so that free-form answers are not truncated.

Judge. Each generation is scored against the reference answer by GEMINI-2.5-FLASH , which sees only the question stem, the reference answer, and the candidate; no video, no audio, no chain-of-thought. The judge returns an integer 0–3 in the MMBench-Video style, and we report binarised accuracy at threshold ≥ 2 . The leaderboard is reproduced in Table 3 of the main text.

F FINE-TUNING RECIPE

F.1 HYPERPARAMETERS

Table 9 summarizes every fine-tuning hyperparameter. The recipe deliberately stays close to a vanilla LLaMA-Factory `qwen2_5_vl` template: LoRA on the seven attention/MLP projections, frozen vision tower, single epoch, cosine schedule. The only non-default choice is the cutoff length, which is raised to 16,384 tokens to fit the long anchor-based prompts at 32 frames.

Eval-time frame ablation. At eval time, the fine-tuned checkpoint is decoded with $\{16, 32, 64\}$ uniformly subsampled frames per question; the headline numbers across these frame budgets and across the four evaluation benchmarks are reported in Table 4.

Table 10: **Annotation quality audit** on the 200-video stratified sample (pipeline annotations) and on 5 fully hand-annotated videos judged under the same rubric.

Annotations	Description precision	Transition precision	Hallucination rate	Audio alignment precision	Entity matching consistency		
					P	R	F1
Pipeline (200 videos)	96.6%	97.2%	2.2%	97.3%	93.1%	97.6%	95.3%
Manual (5 videos)	97.9%	98.1%	0.3%	99.1%	95.4%	99.2%	97.3%

G Annotation Quality Audit

The corpus is annotated automatically, so its reliability has to be measured rather than assumed. We proportionally sampled 200 videos from the corpus by stratified random sampling across all video categories, together spanning more than 50 hours of video, and had human evaluators assess the quality of every annotation channel on them.

Metrics. Five quantities are reported. *Description precision* is the fraction of frame descriptions whose stated content is correct for the frame; for a differential frame only the stated change is judged, since differential frames are not meant to restate the scene. *Transition precision* is the same quantity for the shot-boundary transition descriptions. *Hallucination rate* is the fraction of descriptions that assert content not present in the video at all. *Audio alignment precision* is the fraction of aligned speech segments whose transcript is correct and attached to the correct moment. *Entity matching consistency* scores the cross-shot identity links of the entity bank as precision, recall, and F1: a link between two mentions that denote different entities is a false positive, and two mentions of the same entity left under different identifiers are a false negative.

Results. Table 10 reports the results. To calibrate them, one annotator additionally annotated 5 complete videos by hand, following the same annotation schema, and a second annotator evaluated those manual annotations under the identical rubric. The manual annotations are more precise on every metric, but they are also far sparser: their descriptions are on average 64% shorter and register 37% fewer entities, and producing them took approximately 370× the annotation time of the automatic pipeline. The pipeline therefore trades a small loss of precision for a large gain in coverage, which is the regime a fine-grained, time-resolved corpus needs. The remaining limitations of the automatic annotations lie mainly in incomplete detail and non-exhaustive entity recognition rather than in incorrect statements; the descriptions of the 8B annotator are nevertheless sufficient to construct fine-grained question-answer and reasoning data within our framework.

H Effect of the Annotation Model

To measure how much the choice of Qwen3-VL-8B-Instruct constrains annotation quality, we re-annotated the same 200 audit videos with three larger annotators through the identical pipeline: `Qwen3-VL-30B-A3B` and `Qwen3-VL-32B` from the same model family (Bai et al., 2025), and the proprietary `Gemini-2.5-Flash` (Comanici et al., 2025). Table 11 reports the audit metrics of §G for the four annotators. Larger models yield somewhat more accurate annotations, but for fine-grained annotation the additional gains are small: description and transition precision move by at most a few points, and the largest relative change is in the hallucination rate. Our choice of the 8B annotator reflects a trade-off between annotation cost and annotation quality given the available hardware; the pipeline runs unchanged with any of the four.

I Pipeline Ablations and Hyperparameter Sensitivity

Component ablations. To separate the contributions of the pipeline components, we re-annotated the 200 audit videos with one component switched off at a time and compared the resulting annotations with the shipped ones.

- **Without the entity bank**, contextual descriptions lose consistency and entity names become unstable: the model re-registers the same identity at nearly every cut, so the number of entities per

Table 11: **Audited annotation quality by annotation model** on the 200 audit videos, same rubric as Table 10.

Annotation model	Description precision	Transition precision	Hallucination rate	Entity matching consistency		
				P	R	F1
Qwen3-VL-8B-Instruct (shipped)	96.6%	97.2%	2.2%	93.1%	97.6%	95.3%
Qwen3-VL-30B-A3B	94.1%	97.6%	1.8%	94.7%	98.3%	96.5%
Qwen3-VL-32B	97.9%	98.4%	0.6%	95.7%	98.5%	97.1%
Gemini-2.5-Flash	98.6%	99.3%	0.9%	97.2%	97.4%	97.3%

Table 12: **Re-anchor interval sensitivity** on the 200 audit videos. The shipped value is 300 differential frames.

Re-anchor interval (frames)	**Reference frames per video**
100	90.5 (+1.6%)
300 (shipped)	89.1
600	88.8 (-0.3%)
Never	88.7 (-0.4%)

video rises by +109% , the number of entities that reappear across shots falls by -51% , and the reappearance frequency of an entity falls by -71% .

- **Without the differential mechanism**, every frame is captioned independently and the annotation becomes highly repetitive: the number of entities per video rises by +561.3% and the description length by +97.7% , because every frame restates the scene instead of describing what changed.
- **Without audio**, fewer entities are detected (-10% entities per video) and identity information that is only available in speech is lost (-13% named person entities); for example, a sprinter who is named in the commentary is described only as an athlete in a yellow and black sprinting suit.

The three components thus play distinct roles: the entity bank provides identity consistency across shots, the differential chain keeps the annotation compact and change-oriented, and audio contributes content that the frames alone cannot supply.

Re-anchor interval. The re-anchor interval (300 frames, §B) exists to bound the prompt context inside unusually long shots by promoting the current frame to a new reference frame. Because the number of frames in a single shot of a real video is almost always far below this bound, its value has little effect on the annotation. Table 12 reports the number of reference frames per video on the 200 audit videos for four settings, including disabling the mechanism entirely: no setting moves the count by more than two percent.

J Human Review Statistics and Error Propagation

Verification against the video, not the annotation. Benchmark questions are generated from the annotation stream, so an annotation error can propagate into a wrong question or a wrong answer. The human review stage is designed to catch exactly this: reviewers verify each question against the source video, not against the annotations it was generated from, so a propagated error surfaces as a rejection at this gate. Of 26,363 generated questions, 6,443 pass the five-model text-only audit and 3,011 pass human review; the review therefore rejects 3,432 of the audited questions.

Failure types. Table 13 attributes each rejection to its failure type. Most rejections are factually wrong answers rather than unclear wording; this is the propagation channel that text-only auditing cannot catch and that human review adds. Only 8 of the 6,443 audited questions (0.1%) were judged answerable without the video: after the five-model text-only audit, human reviewers could hardly find any remaining question that did not require watching the video, which is an independent human confirmation of the automatic leakage filter.

The rejection rate is not the annotation error rate. Every retained question has to be visually and temporally grounded, unambiguous, sufficiently challenging, and uniquely answerable, and reviewers are instructed to apply a deliberately conservative standard and to reject a question whenever

Table 13: **Human review outcomes** over the 6,443 text-audited questions, by failure type.

Outcome	Questions	Share of rejections
Accepted (released in KAIROS-Bench)	3,011	—
Rejected: wrong answer	2,860	83.3%
Rejected: question unclear	480	14.0%
Rejected: a distractor is also correct	84	2.5%
Rejected: answerable without the video	8	0.2%

Table 14: **Paired significance of the fine-tuning gains** of Table 4 (Qwen2.5-VL-7B, 32 frames), McNemar's test.

Benchmark	Base	Fine-tuned	p
KAIROS-Bench	40.42	47.49	0.0001
LongVideoBench	56.29	60.08	0.0176
LVBench	38.69	40.81	0.0182
Video-MME	57.11	59.56	0.0031

even one criterion is not fully satisfied. Meeting all of these requirements is particularly hard for long-form videos spanning tens of minutes, where a question can fail because of minor ambiguity, incomplete temporal grounding, or the existence of an alternative valid interpretation. The rejection rate therefore captures the combined difficulty of constructing high-quality long-video questions and the stringency of the review; the error rate of the underlying descriptions is measured directly in §G. The same stringency explains why some capability cells of the released benchmark are small (§M): many questions were generated for them, the review filtered most out, and we did not relax the acceptance criteria for those cells.

Inter-annotator agreement. On a proportional sample of 100 benchmark videos, two additional annotators independently re-screened the questions under the same criteria. Krippendorff's α (Krippendorff, 2011) across the three rounds of screening is 0.82 , indicating substantial agreement and that the screening criteria can be applied consistently.

K Additional Fine-tuning Experiments

K.1 Significance of the Fine-tuning Gains

The base and fine-tuned models answer the same questions, so the gains of Table 4 at the training frame budget are tested with McNemar's test (McNemar, 1947) on the paired per-question outcomes. Table 14 reports the resulting p-values: the differences are statistically significant on all four benchmarks ($p < 0.05$).

K.2 Other Base Models

To test whether the gain depends on the base model, we repeat the fine-tuning experiment of §F with the same KAIROS training data on `InternVL3.5-8B` (Wang et al., 2025b) and `LLaVA-Video-7B` (Li et al., 2024a). Both the base and the fine-tuned models are evaluated with 32 uniformly sampled frames. Table 15 shows that all three base models improve on KAIROS-Bench and on the three external benchmarks after fine-tuning on the KAIROS training set, without any exposure to the training data of those benchmarks.

K.3 An Alternative Training Corpus

We additionally fine-tune `Qwen2.5-VL-7B-Instruct` on LLaVA-Video-178K (Zhang et al., 2024) with the same training configuration and the same number of iterations, and evaluate with 32 sampled frames per video. Table 16 compares the two corpora: KAIROS supervision yields the larger in-domain gain and comparable gains on the external benchmarks.

Table 15: **Fine-tuning on KAIROS training data with three base models.** Accuracy of the fine-tuned model, with the gain over its own base in parentheses; all models use 32 sampled frames.

Base model	KAIROS-Bench	LongVideoBench	LVBench	Video-MME
Qwen2.5-VL-7B-Instruct	47.49 (+7.07)	60.08 (+3.79)	40.81 (+2.12)	59.56 (+2.45)
InternVL3.5-8B	50.52 (+11.90)	58.32 (+4.27)	40.11 (+1.50)	58.46 (+5.24)
LLaVA-Video-7B	35.98 (+9.15)	40.10 (+9.27)	33.86 (+3.30)	39.10 (+7.43)

Table 16: **Two training corpora under the same recipe** (Qwen2.5-VL-7B-Instruct, same configuration and number of iterations, 32 frames). Accuracy with the gain over the base in parentheses.

Training corpus	KAIROS-Bench	LongVideoBench	LVBench	Video-MME
KAIROS	47.49 (+7.07)	60.08 (+3.79)	40.81 (+2.12)	59.56 (+2.45)
LLaVA-Video-178K	45.19 (+4.77)	58.73 (+2.44)	40.78 (+2.09)	58.67 (+1.56)

K.4 NATIVE-VIDEO VERSUS SAMPLED-FRAME INPUTS

The two `Gemini` models and `LLaVA-Video-7B` in Table 2 receive the video natively, whereas every other model receives uniformly sampled frames (Table 8). Comparisons across that boundary should be read with this in mind: the proprietary native-video models see the complete and continuous temporal stream and serve as an upper-bound reference for what native video access affords, rather than as a like-for-like comparison with frame-sampled models.

L DATA GOVERNANCE, LICENSING, AND RELEASE

What is distributed. Following the standard practice of the video understanding community for datasets built on public platform content (Zhou et al., 2018b; Cheng et al., 2025), the public release consists of the source video URLs and our text annotations (per-frame descriptions, transitions, entity banks, aligned speech and environmental audio summaries, and the benchmark questions), licensed under CC BY 4.0; the pixel data of the original videos is not distributed. The complete annotation pipeline is released so that the annotations can be regenerated from the source videos.

Takedown. The ownership of video platforms and creators is respected and applicable data-processing regulations are followed: if the original creator or the platform deletes a video, or a rights holder submits a deletion request, the corresponding URL and all records derived from it are promptly removed from the release.

Representation. The corpus is sourced entirely from two public, large-scale video platforms (9,434 videos from YouTube and 9,570 from Bilibili) and therefore broadly follows the distribution of public long-form web video rather than a balanced geographic or demographic design. Achieving such balance is an ongoing challenge; we document the platform ratio and the geographic and demographic limitations of the current version so that downstream users are aware of potential biases, and the released pipeline allows the community to extend the video pool. Consent and personal-information considerations are discussed in §M.

M LIMITATIONS AND BROADER IMPACT

M.1 LIMITATIONS

VLM bias. All annotations are produced by `Qwen3-VL-8B-Instruct` (Bai et al., 2025), so a systematic perception bias of that VLM also biases KAIROS descriptions; running the pipeline with a different VLM is straightforward and would help quantify the bias.

Language-only entity matching. Cross-shot entity matching is language-only, so for visually-hard-to-describe entities (e.g. visually similar but semantically distinct people in crowd scenes), label-exact matching splits an identity whenever the model paraphrases an established name instead of reusing it, and merges two mentions only when it assigns them the same label. This is addressable

by adding a vision-only re-identification signal as a visual matching stage, which we leave to future work to keep the pipeline VLM-only.

Residual leakage. Even after the multi-model audit and human review, the cross-benchmark leakage probe still finds $+9.6$ pp lift over random under GEMINI-3.1-PRO — the lowest of the ten benchmarks tested but not zero. We attribute this residual to the linguistic anchors that the questions include in lieu of numeric timestamps; removing those anchors would reduce leakage further but would also degrade question grounding for non-visual solvers, so we leave the trade-off explicit rather than tune it.

M.2 BROADER IMPACT

Intended uses. KAIROS is intended for the development and evaluation of long-form video understanding systems in three modes: (i) as supervision data for video-language pretraining and fine-tuning; (ii) as a benchmark for long-form video QA; (iii) as a substrate for evaluating video-language alignment, entity tracking, and temporal grounding methods that go beyond clip-level decisions.

Risks. The dataset contains only annotations of public, platform-distributed videos, with platform-takedown semantics preserved. Personally-identifying information that appears in the source videos (e.g. recognisable individuals in sports broadcasts) is not added by the annotation pipeline; faces and identities are referenced only at the level the source already exposes them. Misuse risks are those typical of any large video corpus; the annotation-side frames are not redistributed, which limits this surface.