

FPS-Bench: A Benchmark for High Frame-Rate Video Understanding

Rohan Choudhury* Jean-Sebastien Dandurand* Kai Qiu Kshitij Madhav Bhat Kartik Sharma
 Liza Dahiya Yizhou Zhao Souraja Kundu Chun-Hsien Lin Kris M. Kitani† László A. Jeni†

Carnegie Mellon University

rchoudhu@andrew.cmu.edu

Abstract

Modern video-language models are typically trained on videos downsampled to low frames-per-second (FPS), and the most commonly used evaluation benchmarks are designed for low-FPS input as well. To address this shortcoming, we present FPS-Bench, a large video question-answering benchmark designed to evaluate VLMs’ capabilities to understand video at high-frame rates. We introduce a new metric, the minimum frames-per-second (minFPS), which measures the minimum frame-rate required to solve a given question. While existing benchmarks require < 1 minFPS, we rigorously curate more than 1000 questions from a diverse source of videos and manually verify minFPS for each example, leading to a benchmark that requires watching videos at on average 7 FPS to solve. Our evaluation of several state-of-the-art VLMs shows that they are severely lacking, achieving QA accuracy of $\approx 30\%$ in the FPS-Bench multiple-choice task, whyeh lieve that FPS-Bench will serve as a valuable tool for improving frontier-level VLMs and release all data and code [here](#).

1. Introduction

Humans seamlessly watch videos at high resolution and at high frame-rate; models do not. The standard practice in today’s state-of-the-art video-language models (VLMs) is to significantly reduce frame-per-second (FPS) of the input video, from 30 to often less than 1 [49]. This design choice persists for two main reasons: (1) videos are extremely large and using a high frame rate requires more input tokens and memory, and (2) a higher FPS often does not always convey more information. Several works have noted that videos tend to have high temporal redundancy [12], and thus many possible video tasks may not require models to perceive videos in the same way that humans do.

This observation is reflected in most video understanding benchmarks. Early benchmarks such as Kinetics [9], ActivityNet [8], or MSR-VTT [55] could be solved with a small number of randomly sampled frames since there was minimal temporal structure to the associated tasks, allowing models to rely on static scene information [32]. More recent and widely used VLM benchmarks such as MVBench [31] and Video-MME [15] also tend to require minimal FPS; the answer for most questions can often be found in a single frame, or the question can be solved while watching the video with as low as 0.1 FPS. This issue persists in long-video understanding benchmarks like EgoSchema [40], and even benchmarks that explicitly focus on fine-grained motion, such as MotionBench [20], can be solved with relatively low FPS.

Given this, what types of question actually *require* models to perceive videos at higher FPS? These questions are not covered by previous benchmarks and, by definition, cannot be solved easily by models that significantly downsample input frames. Although some visual tasks such as object tracking, video object segmentation, and certain aspects of robotic perception require high frame rates, they usually rely on specialized models [25, 37, 45]. However, we aim to identify a set of questions that more general VLMs fail to answer unless the accompanying video is viewed at an adequate frame-rate. For example, consider the video-question pair in Section 1: does the camera flash during the video? Although the camera *does* flash, when viewed at 2 FPS, it is impossible to observe it. This is also true at 4 FPS; only when viewed at 14 FPS can the flash be seen. There is no way to answer the question correctly unless viewed above this FPS; downsampling the video below this will prevent a model from ever answering it correctly.

Based on this idea, we introduce FPS-Bench, an evaluation benchmark for video understanding that consists entirely of such questions. FPS-Bench measures VLM’s ability to answer questions requiring fine-grained temporal reasoning, event localization, and motion sensitivity at high frame rates. We curate 1000 multiple-choice questions an-

*Equal contribution

†Equal advising

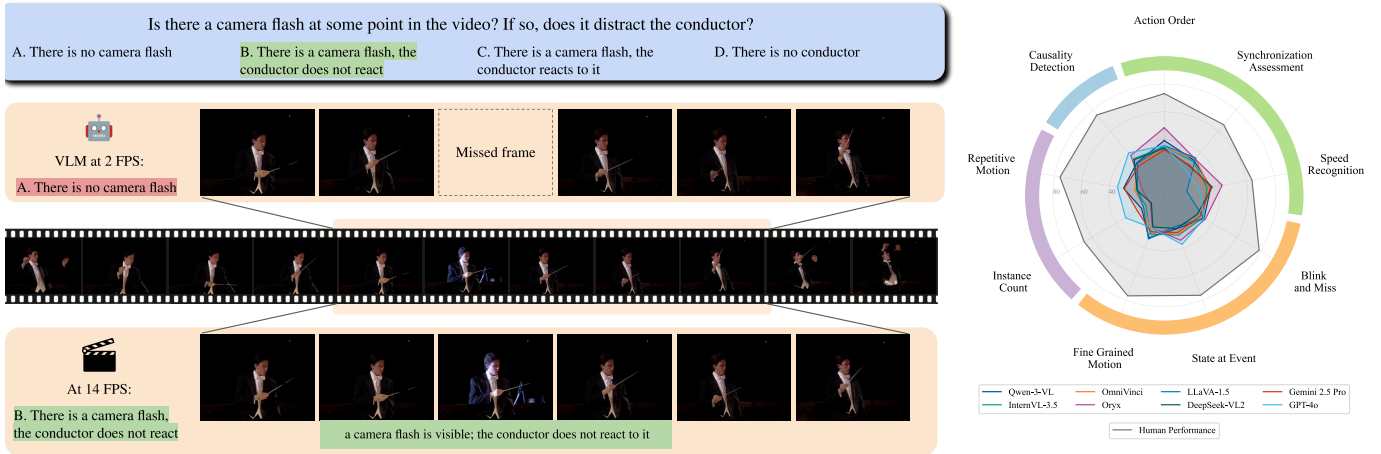


Figure 1. **Analysis of minFPS and VLM Performance on FPS-Bench.** (Left) To correctly answer the question (e.g., identifying the camera flash), a minimum frame rate (here, 14 FPS) is required. Frame rates below this threshold prevent accurate event perception, demonstrating the failure of standard frame downsampling methods. (Right) Comparative performance analysis of current state-of-the-art VLMs against Human Performance on the FPS-Bench. The results are categorized by task type: **Transient Event Perception** (orange), **Temporal Ordering** (light green), **Action Count** (purple), and **Reasoning** (blue).

notated by human experts spanning nine categories, with input videos ranging across a wide variety of visual domains. We manually measure the minimum video FPS required to solve each question, and ensure that every example requires at least 4 FPS, with an average minimum requirement of 6.8 FPS across the dataset.

After curating this high-quality dataset, we evaluated a range of state-of-the-art open- and closed source VLMs and find that they achieve at best 30% accuracy (25% is random), while humans consistently achieve more than 70%. Even when slowing the videos down, the models still perform poorly, suggesting that the failure is not purely due to token limit constraints but rather a fundamental limitation in the models’ ability to perceive, localize, and reason about fine-grained temporal events. In summary, our contributions are as follows.

- We introduce FPS-Bench, the first multiple-choice question-answering benchmark dedicated to evaluating the ability of general VLMs to reason over rapid, high-frequency events across nine distinct temporal categories.
- We propose the concept of Minimum Necessary Frame Rate (minFPS), a metric that quantifies the intrinsic temporal requirement of a video-question pair. We establish the rigor of FPS-Bench with a mandatory threshold of 4 FPS and an average minFPS of 6.8 FPS.
- We provide a comprehensive evaluation of state-of-the-art VLMs on FPS-Bench, revealing that current models struggle significantly with high-frequency temporal reasoning ($\approx 30\%$ accuracy) and highlighting a critical, un-

addressed blind spot in current VLM design.

2. Related Work

2.1. Video-Language Models

Video-language models emerged by extending image-language architectures such as LLaVA [36], BLIP [29] and GPT-4V [2] to handle temporal information. Examples include Video-LLaVA [63] building on LLaVA, VideoChat [30] using BLIP-based encoders, and Qwen2-VL [5] with dynamic temporal sampling, while commercial systems like Gemini [49] provide native video processing and GPT-4o [21] require manual frame extraction. However, due to the large input size of the videos, most models downsample the input heavily to 1 FPS [49], with some [6, 21] using fixed frame counts (e.g. 8-16 frames) regardless of video duration. Although some work aims to mitigate the large context with compression techniques [59, 61], the prevailing choice remains to limit the number of frames through aggressive temporal subsampling. However, some work [7] has found that using higher FPS or resolution consistently improves performance on video benchmarks, and F-16 [34], which trained a video LM at 16 FPS, demonstrated improved performance on fine-grained motion understanding tasks. However, no work has explicitly evaluated the capabilities that are lost as a result of low frame rate sampling.

2.2. Benchmarking Video Understanding

Early video understanding benchmarks focused on activity recognition [16, 23, 24, 26, 48], a classification task that

has been widely found to require very few frames to solve [32]. Larger models pre-trained on large datasets of captions and videos such as HowTo-100M [41], Ego4D [17], or YouTube-8M [1] were then evaluated on video question-answering datasets such as ActivityNet [8, 60], MSR-VTT [55] and MSVD [10]. These require more advanced reasoning than activity recognition, but could still be solved with relatively few frames. Other question-answering benchmarks, such as TVQA [27], Next-QA [53], iVQA [56], suffer from the same issue. More specialized datasets to evaluate compositional reasoning [19], video description ability [54], domain-specific knowledge [46], egocentric video [14, 18] test more specific capabilities, but also require very few frames to solve target tasks. As the size and capacity of the VLM model increased, the field began to focus on understanding more complex videos and advanced reasoning [31]. Long video understanding benchmarks such as EgoSchema [40], Video-MME [15], and LongVideoBench [52] evaluate QA tasks on videos up to an hour long, while fine-grained understanding benchmarks such as PerceptionLM-Bench [11] and MotionBench [20] focus on detailed visual information. Despite requiring more frames overall, these benchmarks can still be solved with sparse temporal sampling (*e.g.*, 1 in every 20 frames for long videos) and do not explicitly test whether models can capture rapid temporal dynamics that necessitate high frame rates.

2.3. High Frame-Rate Video Benchmarks

Understanding videos at higher frame-rate was until recently limited to specialized domains such as object tracking [42, 57], human pose estimation [3, 22], or video object segmentation [43, 45]. Additionally, sports analysis benchmarks such as Sports1M [23], MultiSports [33], FineGym [47], or MMFS [38] often require understanding of high-speed movements, but are still focused on a single domain. More recently, benchmarks such as MotionBench [20], Perception Test [44] evaluated models’ ability to solve a variety of questions with rapid motions; however, they tend to still. Other recent benchmarks such as AirLetters [13] require high FPS, but are still focused on a narrow task. DIVE [62] evaluates VLMs on a toy task that requires recalling subtitles from specific instances, which requires high frame rates but is overly narrow. In contrast, FPS-Bench retains the generality of other benchmarks with its wide range of visual domains and questions, while significantly improving the FPS requirement.

3. Collecting FPSBench

The core principle underlying our paper is that each question should require viewing the accompanying video at relatively high FPS to answer. To formalize this, we introduce the concept of **Minimum Frames-Per Second** (minFPS)

to quantify the temporal requirement for questions in our benchmark. For any given video-question pair, minFPS is defined as the lowest integer frame rate at which a human annotator can consistently reach the correct answer. Crucially, any sampling rate below this threshold must make it impossible to verify the correct answer.

The minFPS metric addresses a limitation of metrics like the temporal certificate, introduced in EgoSchema [40]. The temporal certificate length measures the total duration of the input video a human needs to watch to verify an annotation. However, even though a 10-minute video sampled at 1 FPS and a video sampled at 30 FPS may have the same certificate length (time interval), the 1 FPS video contains far fewer frames, often making rapid high-frequency events indistinguishable. minFPS captures this nuance by focusing on the necessary input FPS; the total number of input frames can be computed by multiplying minFPS and the temporal certificate together.

3.1. Dataset Construction

Measuring minFPS. We determine minFPS through a sequential human verification process: an annotator begins by viewing the input video, question, and answer at 1 FPS. If the annotator is unable to verify the correct answer, they incrementally increase the sampling rate by 1 FPS until the correct answer is unambiguously verifiable. If at 1 FPS, the annotator can already verify the answer, they decrease the FPS by a factor of 2 until this is no longer the case. During this measurement, downsampling is performed by dropping frames, simulating the standard frame sampling process used in modern VLMs. All questions included in FPS-Bench are curated to have a $\text{minFPS} \geq 4$ to ensure high temporal difficulty. We used a threshold of 4 based on our measurements of other benchmarks, which often do not exceed 1 or 2.

Question Types. We define nine distinct question types for FPS-Bench, which focus on temporal reasoning capabilities but also require understanding of rapid, high-frequency motion. The definitions are as follows:

Repetitive Motion: Requires counting the total number of cycles or repetitions of a sustained, high-speed periodic action (*e.g.*, drumstick hits or fan rotations).

Speed Recognition: Requires inferring or comparing the relative speed of an object or action where the difference is only visible at a high frame rate over time.

Fine-Grained Motion: Ascertains subtle or technical differences in movement, form, or trajectory where tiny changes distinguish correct execution (*e.g.*, identifying a specific technique in a sports move).

Action Order: Requires sequencing two or more events that occur nearly simultaneously or in very rapid succession (*e.g.*, determining which object made contact first).

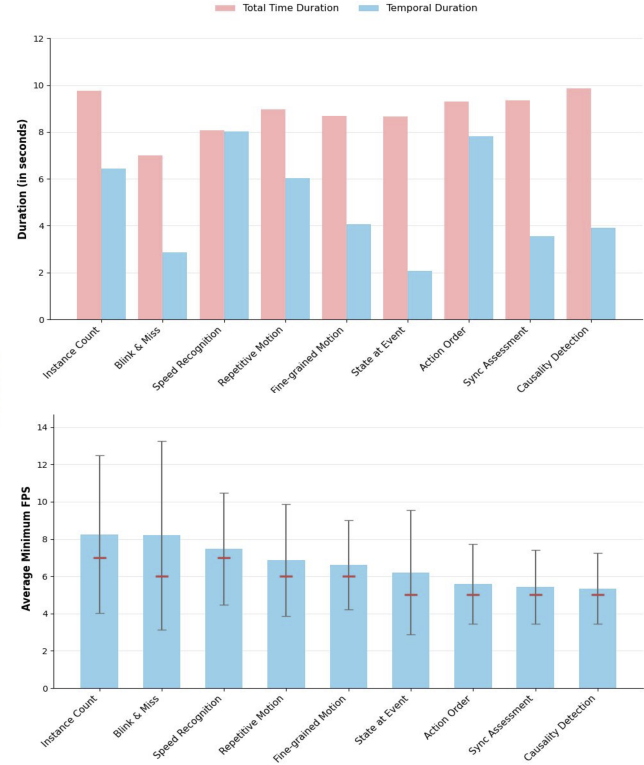


Figure 2. **Dataset Statistics.** FPS-Bench is composed of 1000 questions using 554 videos, drawn from a range of visual domains and uniformly distributed over question categories. Each question is annotated with its minFPS, and all questions have a minFPS of at least 4, with an average of 6.67.

State at Event: Requires identifying the precise state or attribute of an object *at the exact moment* of a rapid, transient interaction.

Blink and Miss: Requires detecting a sudden, non-repetitive, extremely short-duration visual event that may only appear in one or two frames (e.g., spotting a very quick flash or error).

Causality Detection: Identifies the precise temporal relationship between two events, objects, or actions to determine which caused the subsequent event.

Synchronization Assessment: Determines whether two or more independent, high-frequency actions occur at the same time, or whether one occurs before the other.

Instance Count: Calculates the exact number of times a discrete, rapid event occurs within a given time frame (e.g., counting jumps, punches, or quick object passes).

Each of these question types require reasoning about small details, rapid motion, and high-speed events, all of which tend to require higher input FPS. They are also relatively simple for humans while potentially challenging for models, providing a strong evaluation set to highlight capability gaps. We ensure that each category has roughly 110 questions and provide examples for each in the supplement.

Annotation Process. Although several widely used video understanding benchmarks rely on LLM-generated question-answer pairs from captions [31, 40], we opted for entirely human-based annotation due to VLM’s lack of capability to handle our target task. Thus, we recruited several annotators for FPSBench with strong English proficiency and experience with vision-language models. Annotators were tasked with sourcing videos for FPS-Bench from the large and diverse Youtube-8M dataset [1] using the constraints above. YouTube-8M spans an enormous amount of categories and provides high level topic categories to identify videos from. For a given question class, annotators would manually watch several videos from a diverse range of visual domains until they identified a possible question-video pair, then verified whether the question had a minFPS of at least 4. They then created 4 multiple-choice options that were all plausible given the input question and accompanying video, as well as a ‘None of the above’ option for models to express uncertainty.

Quality Review. Following Video-MME and MVBench [15, 31], we used a rigorous review process to ensure high quality annotations. First, each pair of questions and answers was verified by at least two other annotators. The

minFPS for each question was measured by the original labeler and also measured by a different annotator, with the final minFPS set to the minimum of both measurements. A separate annotator also checked that each question was clear and comprehensible, the answer and options coherent and not too obvious, and that each answer was correct given the source video. Finally, three external reviewers answered each question without seeing the answer, and we identified faulty or overly difficult annotations based on questions that the reviewers consistently answered incorrectly. We also used LLMs for quality review, firstly to ensure that all questions had correct spelling and grammar. We then follow the Video-MME protocol [15] to verify that the questions could not be solved without watching the accompanying video. To do this, we remove ‘None of the above’ as an option and run Gemini-1.5 Pro on the questions four times without accompanying video input, randomly shuffled the multiple-choice options each time. Questions in which the LLM found the right answer more than three times were flagged for review and inspected to ensure that there was no clear bias in the question or answer choices and that the video was required.

3.2. Dataset Statistics

We next describe some summary statistics of FPS-Bench, visualized in Figure 2. The benchmark consists of 1000 questions drawn from 554 videos. The questions are categorized into five visual domains: Media & Entertainment, Hobbies & Gaming, Sports & Fitness, Vehicles, and Miscellaneous. For each question, we measured the temporal certificate length for the accompanying video clip, finding that the average certificate length was 2.1 seconds. The benchmark is strictly filtered to a mandatory floor of $\text{minFPS} \geq 4$, with the distribution shown in Figure Z. However, as shown in Figure 2, the questions are often considerably more difficult: the median minFPS is 6.0 FPS and the overall average minFPS is 6.67 FPS. On average, the input videos are approximately 10 seconds long. Figure 2 also shows that each category has a different distribution of minFPS, with blink-and-miss requiring the highest and causality detection the lowest. The high frame rate requirement, applied to relatively short information-dense clips, means that models must process a large number of frames over a short duration and successfully retrieve information from rapid transient events, fundamentally differentiating FPS-Bench from benchmarks that can be solved using simple spatial reasoning or low-frequency temporal patterns.

4. Benchmarking FPSBench

4.1. Comparing minFPS

We first demonstrate that the minFPS from FPS-Bench exceeds that of widely-used datasets. In Fig. 3 we plot the minFPS for a range of video-language

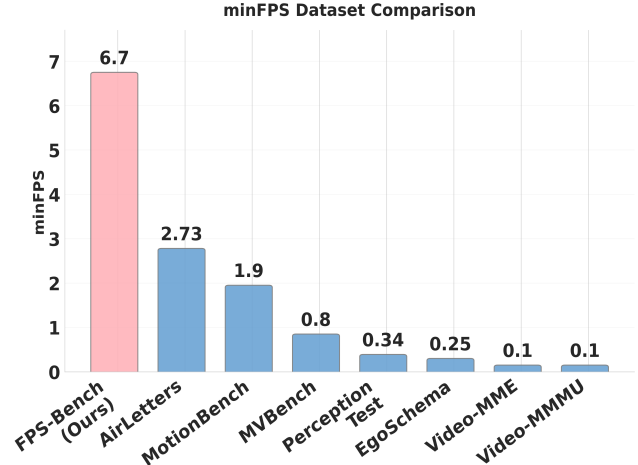


Figure 3. **Comparing minFPS on other datasets.** FPS-Bench has a significantly higher minFPS than other benchmarks, even those aimed at fine-grained motion understanding.

VLM benchmarks, namely (MVBench, PerceptionTest, Video-MMMU, Video-MME, EgoSchema), fine-grained action datasets (MotionBench, AirLetters), and specific temporal reasoning datasets (EgoTempo). Since each of these benchmarks is considerably large, we measure minFPS on a randomly sampled subset of 200 questions in each benchmark. Compared to these, FPS-Bench has a significantly higher minFPS, requiring almost 7 FPS to solve questions on average, while Video-MME requires 0.5. Even on datasets like AirLetters and MotionBench, which are designed for high-speed video understanding, have a minFPS of 2. FPS-Bench exceeds these benchmarks by more than $2.5\times$ without having an overly narrow focus: the question content remains relatively general and spans several diverse visual domains. While benchmarks like EgoSchema and Video-MME test long-video understanding, their minFPS is extremely low, allowing heavy downsampling to reduce the context for longer video inputs.

4.2. Evaluating VLMs on FPS-Bench

We first describe the setup used to evaluate both models and humans and then discuss the results on FPS-Bench.

Benchmarked Models. We benchmark three main classes of models. Firstly, we evaluate two widely used closed-source models that can use video as input, Gemini 2.5 Pro [49] and GPT-4o [21]. Other popular models like Claude [4] and GPT-5 do not accept video input. We also evaluated several open-source video models that span several types of architecture and parameter sizes. These include standard VLMs such as Qwen-3-VL [5, 6, 50], InternVL-3.5 [51], LLaVA-NeXT [63] and LLaVA-OneVision [28]. We also include natively

Models	RM	IC	SR	FM	AO	BM	SE	CD	SY	Overall
Random	25.0%	25.0%	25.0%	25.0%	25.0%	25.0%	25.0%	25.0%	25.0%	25.0%
Qwen-3-VL-8B [6]	18.9%	8.0%	34.9%	29.6%	37.6%	34.0%	26.5%	37.4%	25.9%	28.0%
Qwen-3-VL-32B [6]	29.7%	18.8%	33.9%	32.4%	39.4%	33.0%	25.5%	30.6%	33.0%	30.7%
InternVL-3.5-8B [51]	19.8%	15.2%	31.2%	29.6%	34.9%	32.0%	30.3%	34.3%	32.1%	28.7%
InternVL-3.5-14B [51]	20.7%	14.3%	32.1%	29.6%	42.2%	28.0%	23.2%	30.6%	30.3%	27.9%
OmniVinci [58]	21.6%	17.0%	33.0%	27.8%	33.9%	31.0%	29.6%	29.6%	34.6%	28.5%
LLaVA-OneVision [28]	21.6%	14.3%	30.3%	28.3%	33.9%	33.3%	25.8%	29.6%	26.6%	27.0%
LLaVA-NeXT-Video [28]	16.2%	8.9%	29.4%	28.3%	37.6%	23.0%	35.1%	26.9%	34.9%	26.5%
Oryx [39]	15.3%	11.6%	42.2%	25.0%	48.6%	34.0%	34.3%	37.4%	34.3%	31.3%
LLaVA-1.5 [36]	19.7%	17.2%	16.4%	33.3%	35.8%	33.0%	23.6%	30.0%	35.1%	27.1%
DeepSeek-VL2-Tiny [35]	13.5%	16.1%	24.8%	22.2%	31.2%	27.3%	31.3%	24.3%	32.4%	24.6%
DeepSeek-VL2-Small [35]	19.8%	10.7%	34.9%	24.1%	33.9%	27.3%	25.3%	33.6%	27.8%	26.3%
DeepSeek-VL2-Base [35]	12.6%	7.1%	30.3%	24.1%	32.1%	20.2%	31.3%	28.0%	31.5%	24.0%
Gemini 2.5 Pro [49]	29.1%	22.3%	33.9%	27.8%	32.7%	29.7%	28.3%	28.0%	28.3%	28.9%
GPT-4o [21]	34.2%	32.1%	25.7%	25.0%	35.8%	33.0%	37.4%	39.8%	23.9%	31.8%
Human Performance	76.0%	66.5%	63.9%	76.7%	73.1%	78.6%	76.3%	75.4%	66.3%	72.2%

Table 1. **Main Evaluation Results.** Current state-of-the-art VLMs, whether open-source, closed-source or even image-based, are better than random but perform significantly worse than humans on FPS-Bench.

omni-modal models such as OmniVinci [58] and Oryx [39]. For models with multiple parameter scales such as Qwen or InternVL, we include 8B, 14B and 32B scales if available. Finally, we benchmark open-source image-based models such as DeepSeek-VL-2 [35] and LLaVA [36]. All open-source evaluations used 8 NVIDIA A100 GPUs.

Evaluation Procedure. We follow the protocol of previous work such as Video-MME [15] and use a system prompt that instructs models to focus on fine-grained details to answer the question, as well as strongly encouraging them to output a valid answer choice and textual justification to prevent answers that cannot be scored for accuracy. We include an ablation on prompt design in the supplement, as well as performance with a ‘None-of-the-above’ option.

We next follow the recommended protocol for passing in video for each specific model. Models like Gemini [49] accept raw video mp4 files, while most others require explicit frame sampling. For models that accept raw video, we use their default FPS, while for those that accept frames, we use as many frames as possible. Some models, such as GPT-4o, accept up to 512 frames, allowing us to feed in more than input’s minFPS, while others, such as LLaVA, can only accept up to 64. For these cases, we use the largest possible FPS that covers the entire input clip. We use the native resolution of videos in all cases unless models require resizing.

Human Evaluation. We asked three human experts to answer all the questions from the benchmark and averaged their performance for the human evaluation result in Section 4.1. They watched the input videos on their native FPS (24-30) to answer questions and were presented the multiple choice options in shuffled order and were allowed to

use Google or any relevant resource to look up terms used in the questions that they may have been unfamiliar with. Compared to models, humans score around 70%, more than twice the accuracy of the best-performing model.

Overall Results. The results of our evaluation are in Table 1, which shows a stark gap between state-of-the-art models and humans. Models achieve around 30%, beating random, but performing significantly worse than human testers. This persists across most categories, but models perform particularly poorly in *instance count*. This is plausible, since instance count requires extremely high-FPS understanding while also having very long temporal certificates - the model must often maintain count throughout the entire input clip. On the other hand, the easiest categories are *action order* and *causality detection*, which also have the lowest average minFPS of ≈ 5.3 compared to the rest of the dataset. Surprisingly, we also observe that not all models strictly follow the expected scaling laws. For example, InternVL-14B consistently performs worse than the 8B version, while Qwen-32B outperforms Qwen-8B. Interestingly, while GPT-4o performs very well, the open-source Oryx [39] outperforms Gemini, suggesting the gap between open and closed video understanding models is closing.

4.3. Further Analysis

Temporal Certificates. The results in the previous section establish that a large gap exists between current state-of-the-art VLMs and human performance on FPS-Bench, but the underlying reason is not entirely clear from the table. To better understand where this gap came from, we evaluated the models on two ‘easier’ versions of the benchmark. One potential issue is that, because the questions often rely

Models	Default	Temp. Cert.	Temp. Cert. + minFPS
OmniVinci	28.5%	30.4%	29.1%
LLaVA-OneVision	27.0%	28.8%	28.8%
LLaVA-NeXT-Video	26.5%	26.3%	26.0%
InternVL-3.5-8B	28.7%	29.4%	28.4%
InternVL-3.5-14B	27.9%	28.8%	28.5%
Qwen-3-8B-instruct	28.0%	28.1%	30.6%
Qwen-3-32B-instruct	30.7%	31.6%	33.7%
Oryx	31.3%	30.4%	32.4%
Gemini-2.5-Pro	28.9%	29.7%	33.7%
GPT-4o	31.8%	32.2%	32.1%
Human Performance	72.2%	72.2%	72.2%

Table 2. **Slowing videos down.** When the input videos are slowed down sufficiently for models to guarantee seeing the key frames, their performance improves, but is not close to human level.

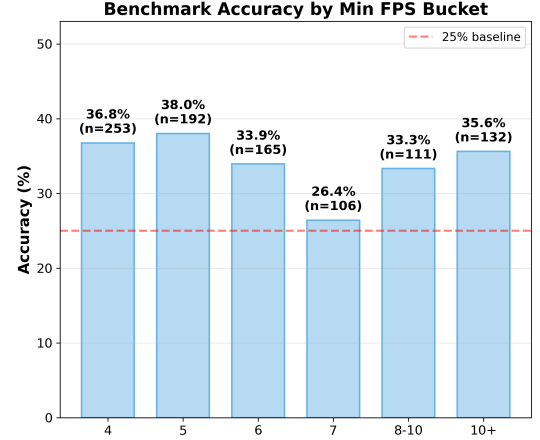


Figure 4. **GPT4o: Accuracy vs FPS.** When comparing the overall FPS to accuracy, we find that GPT-4o performs worst on questions requiring around 7 FPS, rather than steadily decreasing.

Number of Frames							
Models	#Param.	16	32	64	128	256	512
OmniVinci	8B	30.80%	31.40%	31.60%	32.60%	32.60%	30.90%
LLaVA-OneVision	7B	31.50%	29.60%	30.50%	30.30%	29.50%	27.20%
LLaVA-NeXT-Video	34B	26.60%	27.90%	28.00%	26.90%	28.10%	28.00%
InternVL-3.5-8B	8B	33.10%	32.80%	32.80%	29.50%	26.70%	26.50%
InternVL-3.5-14B	14B	29.20%	28.00%	30.80%	29.40%	26.50%	26.40%
FPS							
Models	#Param.	1	2	4	8	16	30
Qwen-3-8B-instruct	8B	27.60%	28.20%	30.10%	30.60%	33.60%	34.10%
Qwen-3-32B-instruct	32B	27.10%	29.8%	32.40%	33.00%	35.10%	35.20%
Oryx	34B	29.40%	29.60%	31.40%	30.50%	31.0%	30.50%
InternVL-3.5-8B	8B	30.60%	33.00%	32.10%	31.60%	29.20%	28.90%
InternVL-3.5-14B	14B	30.40%	29.40%	31.40%	30.00%	26.90%	26.70%
Human Performance	-	-	-	-	-	-	72.2%

Table 3. **Varying input frames and FPS.** While some models improve with more frames or higher FPS, other models start to perform worse, suggesting issues with handling long context. Each model’s best accuracy is bolded.

on a single, hard-to-see moment, the model is struggling with retrieval rather than reasoning. We test this hypothesis by passing only the temporal certificate for each question instead of the full 10-second interval. As shown in Figure 4, this does not improve performance; removing the retrieval problem does not make the problem easier for the model.

Slowing Videos Down. To make the problem even easier and more fair for different models, we slow each input video down so that the models ‘perceive’ the video in the question’s ground-truth minFPS. For example, Gemini Pro slows input videos down to 1 FPS [46], so we set each input video to be slower by a factor of its minFPS. A question that has minFPS of 5 would be slowed down by a factor of 5, so that sampling at 1 FPS would yield all the necessary frames. For models like GPT-4o or Qwen-VL that receive

a fixed number of frames, we sample at the target minFPS. Table 2 shows that this procedure improves performance but does not close the gap with human performance. This suggests that even if models have access to more of the relevant context, they are unable to consistently reason with it.

Varying FPS. Next, we measure the accuracy of the models as the input FPS increases. As shown in Table 3, we measured the performance of open-source models as we increased the number of input frames or the input FPS. Some models, such as Qwen-3-VL, continuously improve as FPS increases. However, most model’s performance does not significantly change, supporting the observation that there is a lack of capability on fine-grained reasoning tasks. We further speculate that as FPS continues to increase, the input context continues to grow, with relatively small differences

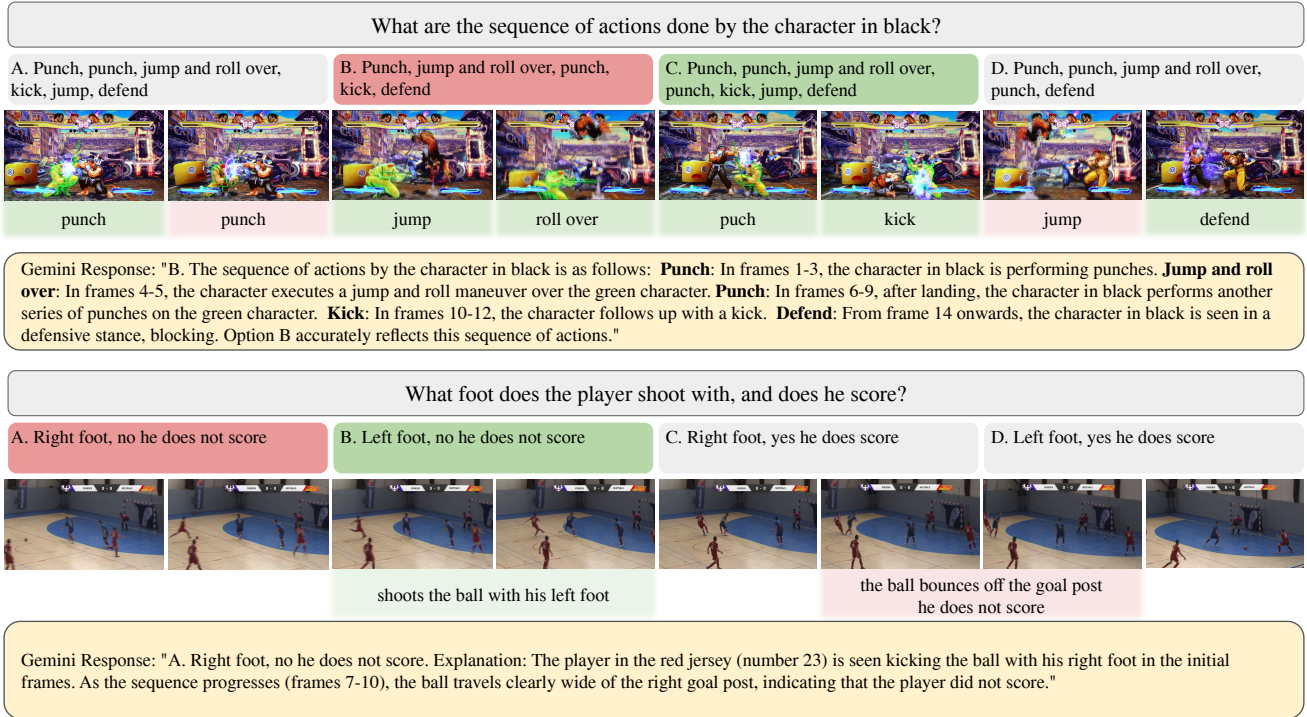


Figure 5. **Failure Cases.** We provide two representative failures from Gemini showing how it predicts the incorrect answer even when provided all of the necessary input frames. In both cases, Gemini misses small details that occur quickly and are key to obtaining the correct answer.

between frames. Most VLMs are unlikely to have trained on a significant number of such situations and thus currently lack the capabilities to make the most out of extra frames.

Qualitative Analysis. We analyze Gemini’s incorrect responses on two representative examples in Figure 5 to demonstrate common failure modes. For both examples, we slow down the video so that Gemini can see the required frames. The top example is an *action order* question that requires getting the order of actions correct. Not only is the order important, but viewing the video at full FPS is required to see every action in the sequence. Although the video is slowed so that Gemini sees the entire sequence, it fails to understand two of the actions in the sequence and chooses B instead of C. To answer the second question, a model must see both that the player very quickly kicks a small ball, and that the ball bounces off the post towards the end. Even when provided with all the necessary frames, Gemini misses key points: it correctly identifies that the player does not score but misses the rapid motion of the shot, resulting in an incorrect response. These two examples suggest that models still have significant room to improve in understanding fine-grained details in video.

5. Discussion

Limitations. One limitation of FPS-Bench is that the visual domains and types of question are biased towards understanding and answering high-speed short-term events. This may not reflect the majority of current VLM users’ use-cases, which other benchmarks already cover. Secondly, compared to other video benchmarks, FPS-Bench is relatively small. However, the annotation pipeline we described is relatively scalable, and we anticipate scaling up this benchmark in the near future as models adopt it.

In summary, we introduced FPS-Bench, a new video question-answering benchmark that focuses on questions that require watching videos *at high FPS*. To perform well on FPS-Bench, a model must be able to accept a large number of frames and understand extremely rapid and quick motions. We collect FPS-Bench with 1000 such questions and manually verify that each question met the benchmark constraints. Evaluation results showed that no current state-of-the-art model comes close to human performance on FPS-Bench, even when given slowed-down versions of the video, demonstrating an underlying capabilities gap. We believe that FPS-Bench will play a key role in the development of video understanding going forward.

Acknowledgments

This work was partially supported by Fujitsu Research of America. RC is supported by the NSF GRFP.

References

- [1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*, 2016. 3, 4
- [2] 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. 2
- [3] Mykhaylo Andriluka, Umar Iqbal, A Milan, E Insafutdinov, L Pishchulin, Juergen Gall, and Bernt Schiele. Posetrack: A benchmark for human pose estimation and tracking. *arXiv preprint arXiv:1710.10000*, 2017. 3
- [4] Anthropic. Introducing the next generation of claude, 2024. 5
- [5] 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. 2, 5
- [6] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 2, 5, 6
- [7] Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, et al. Paligemma: A versatile 3b vlm for transfer. *arXiv preprint arXiv:2407.07726*, 2024. 2
- [8] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 961–970, 2015. 1, 3
- [9] Joao Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A short note about kinetics-600. *arXiv preprint arXiv:1808.01340*, 2018. 1
- [10] David Chen and William B Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pages 190–200, 2011. 3
- [11] Jang Hyun Cho, Andrea Madotto, Effrosyni Mavroudi, Triantafyllos Afouras, Tushar Nagarajan, Muhammad Maaz, Yale Song, Tengyu Ma, Shuming Hu, Suyog Jain, et al. Perceptionlm: Open-access data and models for detailed visual understanding. *arXiv preprint arXiv:2504.13180*, 2025. 3
- [12] Rohan Choudhury, Guanglei Zhu, Sihan Liu, Koichiro Niinuma, Kris Kitani, and László Jeni. Don’t look twice: Faster video transformers with run-length tokenization. *Advances in Neural Information Processing Systems*, 37:28127–28149, 2024. 1
- [13] Rishit Dagli, Guillaume Berger, Joanna Materzynska, Ingo Bax, and Roland Memisevic. Airletters: An open video dataset of characters drawn in the air. *arXiv preprint arXiv:2410.02921*, 2024. 3
- [14] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. The epic-kitchens dataset: Collection, challenges and baselines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):4125–4141, 2020. 3
- [15] Chaoyou Fu, Yuhao 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 Computer Vision and Pattern Recognition Conference*, pages 24108–24118, 2025. 1, 3, 4, 5, 6
- [16] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The” something something” video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*, pages 5842–5850, 2017. 2
- [17] 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*, pages 18995–19012, 2022. 3
- [18] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19383–19400, 2024. 3
- [19] Madeleine Grunze-McLaughlin, Ranjay Krishna, and Maneesh Agrawala. Agqa: A benchmark for compositional spatio-temporal reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 3
- [20] Wenyi Hong, Yean Cheng, Zhuoyi Yang, Weihao Wang, Lefan Wang, Xiaotao Gu, Shiyu Huang, Yuxiao Dong, and Jie Tang. Motionbench: Benchmarking and improving fine-grained video motion understanding for vision language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 8450–8460, 2025. 1, 3
- [21] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 2, 5, 6
- [22] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments.

- IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, 2014. 3
- [23] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014. 2, 3
- [24] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 2
- [25] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024. 1
- [26] Hildegard Kuehne, Hueihan Jhuang, Estfbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *2011 International conference on computer vision*, pages 2556–2563. IEEE, 2011. 2
- [27] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara Berg. Tvqa: Localized, compositional video question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 1369–1379, 2018. 3
- [28] 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*, 2024. 5, 6
- [29] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022. 2
- [30] KunChang Li, Yinan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and Yu Qiao. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023. 2
- [31] 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*, pages 22195–22206, 2024. 1, 3, 4
- [32] Yingwei Li, Yi Li, and Nuno Vasconcelos. Resound: Towards action recognition without representation bias. In *Proceedings of the European conference on computer vision (ECCV)*, pages 513–528, 2018. 1, 3
- [33] Yixuan Li, Lei Chen, Runyu He, Zhenzhi Wang, Gangshan Wu, and Limin Wang. Multisports: A multi-person video dataset of spatio-temporally localized sports actions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 13536–13545, 2021. 3
- [34] Yixuan Li, Changli Tang, Jimin Zhuang, Yudong Yang, Guangzhi Sun, Wei Li, Zejun Ma, and Chao Zhang. Improving llm video understanding with 16 frames per second. *arXiv preprint arXiv:2503.13956*, 2025. 2
- [35] Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Daya Guo, et al. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*, 2024. 6
- [36] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 2, 6
- [37] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pages 38–55. Springer, 2024. 1
- [38] Sheng-Lan Liu, Yu-Ning Ding, Gang Yan, Si-Fan Zhang, Jin-Rong Zhang, Wen-Yue Chen, and Xue-Hai Xu. Fine-grained action analysis: A multi-modality and multi-task dataset of figure skating. *arXiv preprint arXiv:2307.02730*, 2023. 3
- [39] Zuyan Liu, Yuhao Dong, Ziwei Liu, Winston Hu, Jiwen Lu, and Yongming Rao. Oryx mllm: On-demand spatial-temporal understanding at arbitrary resolution. *arXiv preprint arXiv:2409.12961*, 2024. 6
- [40] Kartikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. *Advances in Neural Information Processing Systems*, 36:46212–46244, 2023. 1, 3, 4
- [41] 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 *ICCV*, 2019. 3
- [42] Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016. 3
- [43] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017. 3
- [44] Viorica Pătrăucean, Lucas Smaira, Ankush Gupta, Adrià Recasens Contiente, Larisa Markeeva, Dylan Banarse, Skanda Koppula, Joseph Heyward, Mateusz Malinowski, Yi Yang, Carl Doersch, Tatiana Matejovicova, Yury Sulsky, Antoine Miech, Alex Frechette, Hanna Klimczak, Raphael Koster, Junlin Zhang, Stephanie Winkler, Yusuf Aytar, Simon Osindero, Dima Damen, Andrew Zisserman, and João Carreira. Perception test: A diagnostic benchmark for multimodal video models. In *Advances in Neural Information Processing Systems*, 2023. 3
- [45] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 1, 3
- [46] Mohammadreza Reza Salehi, Jae Sung Park, Aditya Kupati, Ranjay Krishna, Yejin Choi, Hanna Hajishirzi, and Ali

- Farhadi. Actionatlas: A videoqa benchmark for domain-specialized action recognition. *Advances in Neural Information Processing Systems*, 37:137372–137402, 2024. 3, 7
- [47] Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. Finegym: A hierarchical video dataset for fine-grained action understanding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3
- [48] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 2
- [49] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 1, 2, 5, 6
- [50] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 5
- [51] 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*, 2025. 5, 6
- [52] 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. 3
- [53] 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*, pages 9777–9786, 2021. 3
- [54] Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. Video question answering via gradually refined attention over appearance and motion. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1645–1653, 2017. 3
- [55] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5288–5296, 2016. 1, 3
- [56] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Just ask: Learning to answer questions from millions of narrated videos. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1686–1697, 2021. 3
- [57] Jinyu Yang, Mingqi Gao, Zhe Li, Shang Gao, Fangjing Wang, and Feng Zheng. Track anything: Segment anything meets videos, 2023. 3
- [58] Hanrong Ye, Chao-Han Huck Yang, Arushi Goel, Wei Huang, Ligeng Zhu, Yuanhang Su, Sean Lin, An-Chieh Cheng, Zhen Wan, Jinchuan Tian, et al. Omnivinci: Enhancing architecture and data for omni-modal understanding llm. *arXiv preprint arXiv:2510.15870*, 2025. 6
- [59] Jiabo Ye, Haiyang Xu, Haowei Liu, Anwen Hu, Ming Yan, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. mplug-owl3: Towards long image-sequence understanding in multi-modal large language models. *arXiv preprint arXiv:2408.04840*, 2024. 2
- [60] 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*, pages 9127–9134, 2019. 3
- [61] Boqiang Zhang, Kehan Li, Zesen Cheng, Zhiqiang Hu, Yuqian Yuan, Guanzheng Chen, Sicong Leng, Yuming Jiang, Hang Zhang, Xin Li, et al. Videollama 3: Frontier multimodal foundation models for image and video understanding. *arXiv preprint arXiv:2501.13106*, 2025. 2
- [62] Haichao Zhang, Wenhao Chai, Shwai He, Ang Li, and Yun Fu. Dense video understanding with gated residual tokenization. *arXiv preprint arXiv:2509.14199*, 2025. 3
- [63] Yuanhan Zhang, Bo Li, haotian Liu, Yong jae Lee, Liangke Gui, Di Fu, Jiashi Feng, Ziwei Liu, and Chunyuan Li. Llava-next: A strong zero-shot video understanding model, 2024. 2, 5