

LiveCC: Learning Video LLM with Streaming Speech Transcription at Scale

Joya Chen^{1*} Ziyun Zeng^{1*} Yiqi Lin^{1*} Wei Li² Zejun Ma² Mike Zheng Shou^{1✉}

¹Show Lab, National University of Singapore ²ByteDance



Figure 1. LiveCC provides real-time commentary for streaming video, emulating a human commentator. This example is drawn from the YouTube video (ID: [I7pTpMjqNRM](#)), featuring the Paris 2024 Olympics Men’s Basketball Final between France and the USA. Our 7B model generates continuous commentary with a latency of less than 0.5 seconds per frame, supporting real-time applications at 2 FPS.

Abstract

Recent video large language models (Video LLMs) often depend on costly human annotations or proprietary model APIs (e.g., GPT-4o) to produce training data, which limits their training at scale. In this paper, we explore large-scale training for Video LLM with cheap automatic speech recognition (ASR) transcripts. Specifically, we propose a novel streaming training approach that densely interleaves the ASR words and video frames according to their timestamps. Compared to previous studies in vision-language representation with ASR, our method naturally fits the streaming characteristics of ASR, thus enabling the model to learn temporally-aligned, fine-grained vision-language modeling. To support the training algorithm, we introduce a data production pipeline to process YouTube videos and their closed captions (CC, same as ASR), resulting in [Live-CC-5M](#) dataset for pre-training and [Live-WhisperX-526K](#) dataset for high-quality supervised fine-tuning (SFT). Remarkably, even

without SFT, the ASR-only pre-trained [LiveCC-7B-Base](#) model demonstrates competitive general video QA performance and exhibits a new capability in real-time video commentary. To evaluate this, we carefully design a new [LiveSports-3K](#) benchmark¹, using LLM-as-a-judge to measure the free-form commentary. Experiments show our final [LiveCC-7B-Instruct](#) model can surpass advanced 72B models (Qwen2.5-VL-72B-Instruct, LLaVA-Video-72B) in commentary quality even working in a real-time mode. Meanwhile, it achieves state-of-the-art results at the 7B/8B scale on popular video QA benchmarks such as VideoMME and OVObench, demonstrating the broad generalizability of our approach. All resources of this paper have been released at [showlab.github.io/livecc](#).

1. Introduction

The success of large language models (LLMs) [3, 4, 22, 62, 63, 78, 79, 91] owes much to the large-scale auto-regressive

✉Corresponding Author. *Equal Contribution.

¹Welcome to participate the real-time video commentary competition at CVPR25 workshop [loveucvpr25/track2](#). Free GPT-4o judging provided.

language pre-training [8, 31, 36, 69, 70]. This inspires large multimodal models (LMMs) [5, 29, 54, 64, 112], which are initially only achieved by small-scale instruction tuning [54, 112], to increasingly emphasize data scaling during their training. While early large multimodal models (LMMs) such as LLaVA [54] were only supervised fine-tuned on 158K image QA samples, recent advanced approaches [16, 42, 50, 59, 105] have expanded the training data to millions of multimodal conversation samples, substantially benefiting from the increased data size.

A long-term ambition in this field is to develop LMMs akin to *J.A.R.V.I.S.*, seamlessly assisting humans in real-life scenarios. Building on prior successes in LLMs/LMMs, a possible way is to collect extensive streaming video-text chat data. For instance, recordings of a basketball coach providing real-time feedback to a novice player could be great data for training. However, previous studies on streaming video LLMs [13, 24, 25, 55, 67, 68, 82, 83, 86, 98, 100, 111] have explored the difficulties of collecting and scaling up such data. They either rely on LLMs to generate “hallucinated” streaming conversations from video annotations, or fine-tune on small-scale dense caption datasets [32, 37, 110]. Neither approach is scalable enough to yield a truly powerful streaming video LLM.

To address these limitations, two primary approaches merit consideration. First, recent video-text datasets [14, 15, 81, 105] increasingly employ advanced LMMs, such as GPT-4o [29], for synthetic data generation. While effective, this approach is costly and risks violating usage terms. Another alternative leverages the inherent audio channel in videos by utilizing automatic speech recognition (ASR) transcriptions as textual data. Prior works [60, 88–90, 96] have explored large-scale video-ASR learning but typically treat ASR transcriptions as global video captions, overlooking their valuable temporal alignment. In practice, some ASR texts naturally synchronize with visual content, offering an untapped opportunity for video-language learning, especially for streaming applications.

In this work, we aim to scale video LLM training by ASR transcriptions. We propose a novel streaming training approach that densely interleaves ASR words with corresponding video frames, as illustrated in Figure 5. The model is trained to generate frame assigned ASR words in an autoregressive manner. This approach marks a significant departure from prior LMMs [5, 13, 54, 111, 112], which primarily learn from complete sentences or paragraphs. In contrast, our method simply learns the native *short, incomplete* ASR word sequences that are temporally aligned with video frames. This offers three key advantages: 1) it aligns naturally with the real-world data, making it readily applicable to video platforms like YouTube; 2) it enables the model to learn fine-grained temporal correlations between visual content and spoken language; and 3) during inference, it fa-

cilitates seamless streaming by generating only a few words per frame, ensuring extremely low latency.

To achieve this goal, we address three fundamental challenges: 1) How can video-ASR data be effectively curated and selected for training? 2) How should video-ASR streaming sequences be efficiently modeled? 3) How can streaming word generation—termed real-time video commentary—be rigorously evaluated? To tackle these challenges, we first design a data collection pipeline that integrates cost-effective techniques to enhance ASR quality and improve visual-text alignment, such as active speaker detection [47] for filtering low-quality talking-head videos. This pipeline enables the construction of the Live-CC-5M pre-training set and the Live-WhisperX-526K SFT set. Next, we incorporate our streaming pre-training approach into the Qwen2-VL-7B-Base [80] base model, yielding LiveCC-7B-Base, and investigate key factors influencing accurate ASR word prediction, such as leveraging video title and previous ASR as context to mitigate the learning ambiguity. Then, we introduce LiveSports-3K, a new benchmark that employs the LLM-as-a-judge [109] framework for evaluating real-time video commentary. We fine-tune LiveCC-7B on Live-WhisperX-526K and LLaVA-Video-178K [105] to obtain LiveCC-7B-Instruct, achieving state-of-the-art performance on general QA and streaming commentary tasks.

Extensive experiments demonstrate that our streaming pre-training approach on Live-CC-5M substantially enhances commentary quality and yields improvements in general video QA performance. By fine-tuning our pre-trained model using the Live-WhisperX-526K dataset in conjunction with LLaVA-Video-178K, our method achieves state-of-the-art results on popular video QA benchmarks such as Video-MME [23] and OVOBench [46], as well as our proposed LiveSports-3K benchmark, and delivers competitive performance on MVBench [45]. These results indicate that our comprehensive framework is not only for real-time video commentary but also beneficial to common video understanding capability.

2. Related Work

Large Multimodal Models. Early LMMs [2, 20, 43, 54, 112] achieve image dialogue by projecting the visual embedding (*e.g.*, from CLIP [71, 97]) to align with LLM token embedding space. Then, lots of efforts explore more free-form interleaved vision-text chatting [3, 5, 40, 42, 50, 53, 64, 93], spatial/temporal grounding [12, 38, 52, 74, 80, 95, 104], video comprehension [24, 27, 29, 44, 48, 57, 75, 77, 90, 99, 106], etc. Our model is also an LMM, but it offers new insights into cost-effective and scalable ASR training data, as well as a new capability of real-time video commentary.

Training Video LLMs. Popular video LLMs [6, 16, 80, 98, 106] typically rely on human- or LLM-crafted video

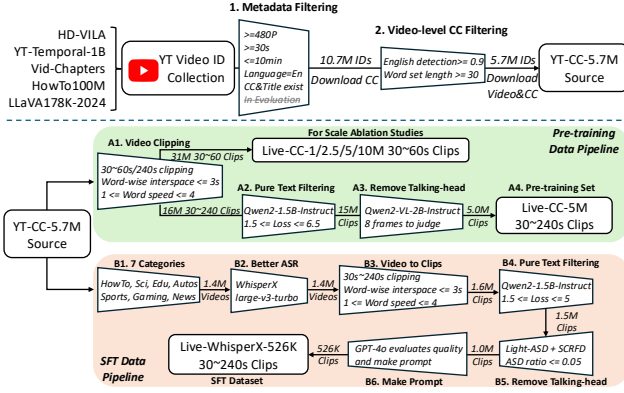


Figure 2. LiveCC data production pipeline. We begin by integrating several large-scale YouTube video datasets [60, 88, 89, 96, 105], followed by metadata filtering, resulting in a curated pool of 5.7M videos. Then, the pre-training dataset is built using the original YouTube CC, while the SFT dataset leverages higher-quality ASR transcriptions generated by WhisperX [7, 72]. We also introduce a set of efficient filtering techniques to improve the SFT data quality. Please refer to Section 3.1 for details.

caption/QA sequences for training. In contrast, our work focuses on training less explored ASR transcription data, leveraging its scalability and automatic extraction capabilities. Several studies [51, 60, 88–90, 96] have investigated learning spatio-temporal representations through video-ASR pre-training. The most related work is Vid2Seq [90], which pre-trains a model to predict timestamped ASR paragraph for videos. However, its training paradigm still aligns with previous video captioning, aiming to predict an overall event. In contrast, our approach aligns with the streaming ASR, learning to predict short, incomplete ASR words per frame causally, thereby enabling more fine-grained spatial-temporal learning.

Streaming Video Understanding. Traditional video understanding benchmarks [1, 10, 11, 28, 30, 34, 39, 58, 85] allow models to access entire video frames before making predictions, a setting commonly referred to as “offline”. However, this paradigm does not align well with many real-time applications (e.g., AR glasses). Previous online video understanding tasks, such as online action detection [26, 107, 108], localization [9, 35, 76], and captioning [111], primarily focus on densely identifying current or future actions. Recent advancements in streaming video LLMs [13, 24, 25, 55, 67, 68, 82, 83, 98, 111] and benchmarks [33, 46, 49, 86] have introduced capabilities such as proactive response, long-form streaming, and interactive multimodal conversation. However, they heavily rely on manual or GPT crafted data, and will be “blind” for video input before the text/audio generation finished. Our work provides a comprehensive solution that leverages ASR data to enhance both general QA and streaming capabilities, and

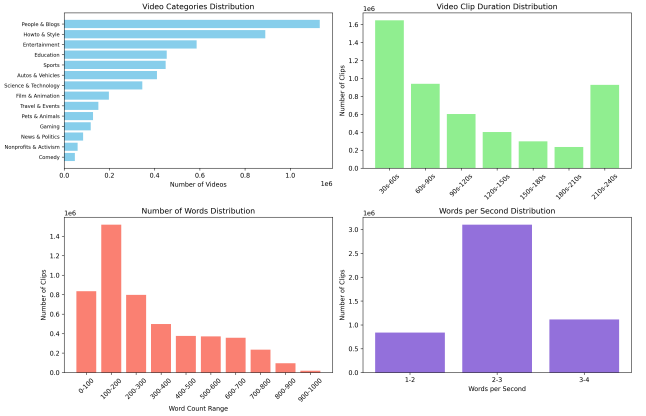


Figure 3. Overview of our proposed Live-CC-5M dataset.

achieves novel real-time video commentary feature, along with a new benchmark LiveSports-3K-CC for evaluation.

3. Methodology

3.1. Video-ASR Data Curation

To demonstrate the scalability of our pre-training strategy, we aggregate four recent large-scale video datasets—HD-VILA [88], YT-Temporal-1B [96], VidChapters [89], and HowTo100M [60]—as our video sources. As illustrated in Figure 2, we begin by retrieving video metadata (e.g., title, duration, category) and the corresponding YouTube closed captions (CC) using the released video IDs. To ensure high visual quality, we retain only videos with a resolution of at least 480p. For storage efficiency, we filter videos to be between 30 seconds and 10 minutes in length. We further require the presence of both CC and title metadata. Additionally, we observe that English videos typically yield better ASR quality; therefore, we restrict our selection to English-language content. Applying these filtering criteria results in a curated set of 10.7 million YouTube video IDs.

YT-CC-Source-5.7M. We further observed that YouTube metadata often mislabels the language category, e.g., marking videos as English despite containing code-mixed content or garbled characters. To address this, we apply the XLM-RoBERTa [19] (*papluca/xlm-roberta-base-language-detection*) for English detection, using a confidence threshold of 0.9. In addition, we discard video IDs with sparse CC, e.g. music videos with only a few words. We require each video to contain at least 30 distinct words in its CC. Applying these filters, we download these 5.7 million videos with English CC, which serves as the source for both pre-training and SFT datasets.

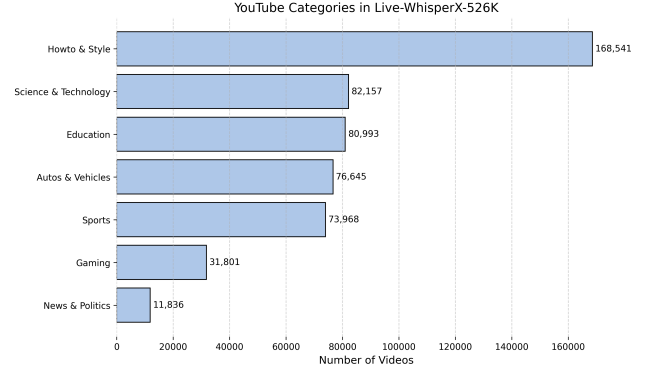
YT-CC-Source-5.7M→Live-CC-5M. Upon inspection, we observed that YouTube CC is generally of low quality—lacking punctuation, case-insensitive, and frequently containing garbled characters. Nevertheless, due to their accessibility and low cost, they offer a scalable data source,

making them more suitable for pre-training rather than SFT.

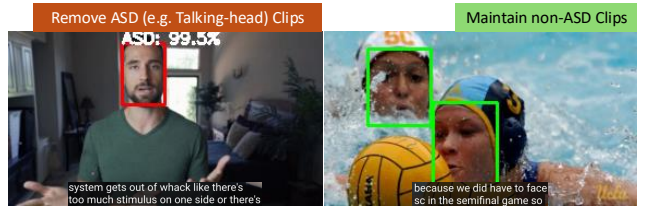
Therefore, we design the following steps for data curation: **A1)** First, we segment the video based on ASR word timestamp gaps. If the gap between words exceeds 3 seconds, a new clip is generated. If a clip exceeds the maximum length, it is split into a new clip. Clips shorter than 30 seconds or with a speech rate outside the 1 to 4 words per second range are discarded. The extended range of silence or abnormal speech speed makes it hard for the model to learn the end-of-sequence (EOS) predictions. For pre-training, the default maximum clip length is set to 240 seconds. For ablation studies, the clip length is set to 60 seconds for training efficiency. We rank these clips by their word set size, which reflects content informativeness, and create pre-training subsets with 1M, 2.5M, 5M, and 10M clips. **A2)** We compute the pure text loss of ASR transcripts by language model to assess their dependency on visual content. A very low perplexity suggests the transcript is self-contained and does not require visual grounding, while a very high perplexity often correlates with poor ASR quality. Empirically, we use Qwen2-1.5B-Instruct [91] to retain samples with loss values in the range of 1.5 to 6.5; **A3)** To remove videos with people consistently facing the camera and talking without meaningful visual information, we apply visual filtering using Qwen-VL-2B-Instruct [80]. For each video, we use 8 uniformly sampled frames to detect persistent face-speaking content by prompting Qwen-VL-2B-Instruct. We keep the videos if the model’s confidence in detecting the talking head is below a threshold of 0.3.

Finally, we obtain Live-CC-5M for pretraining. Figure 3 shows the statistics of these data samples.

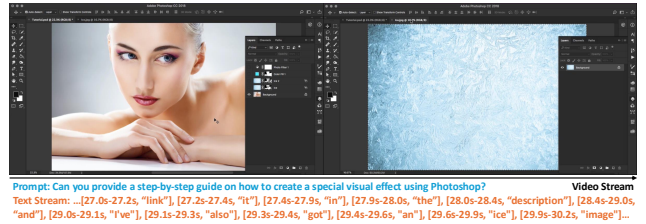
YT-CC-Source-5.7M→Live-WhisperX-526K. As the low-quality YouTube CC makes them unsuitable for SFT data, we further perform the following steps to obtain high-quality, visually grounded ASR transcription: **B1)** We only maintain 7 YouTube categories shown in Figure 4a but filter out “People & Blogs” and “Film & Animation”, as their ASR content typically lacks correspondence with the visual events; **B2)** We employ WhisperX [7, 72] (large-v3-turbo) to generate more accurate, word-level aligned ASR transcriptions; **B3)** Similar to Step A1, the maximum clip length is 240 seconds. For pretraining, since pre-ASR provides context, clips can be split in the middle of sentences. However, during the instruction fine-tuning stage, where no pre-ASR context is available, we ensure that each clip begins at the start of a sentence. Specifically, the last ASR word must be a period, question mark, or exclamation mark, and the current clip must start with a capital letter; **B4)** The same as Step A2, while the range of text perplexity is 1.5 to 5; **B5)** Despite the above filtering steps like Step A3, we observe that many remaining videos are dominated by talking-head content, which is often useless for training real-time video



(a) Statistics of our proposed Live-WhisperX-526K dataset.



(b) An example of ASD removal in SFT data pipeline.



(c) An example from the Live-WhisperX-526K dataset.

Figure 4. Overview of the Live-WhisperX-526K dataset.

commentary. To address this, we employ active speaker detection (ASD) [47] to identify and exclude such videos. For efficiency, we optimize Light-ASD [47] pipeline in face detection, tracking, and multiprocessing, achieving a $250\times$ speed-up. As a result, processing a 5-minute video now takes only 1–1.5 seconds. An ASD removal example is in Figure 4b. **B6)** Since these ASR transcripts lack associated user prompts, we employ GPT-4o [29] to generate a prompt for each sample. The prompts are crafted to match the style and intent of the speech transcription without revealing specific content. With this prompt, we no longer need pre-ASR applied during SFT.

Finally, we get a high-quality SFT dataset comprising 526K video clips, each paired with word-level timestamped ASR transcripts and a user prompt. Figure 4c shows an example in Live-WhisperX-526K dataset.

3.2. Modeling

Training with Dense Interleaving Sequence. As shown in Figure 5, our model architecture builds upon Qwen2-VL [80], which integrates a Vision Transformer [21] with basic dynamic resolution support and uses Qwen2 [91] as

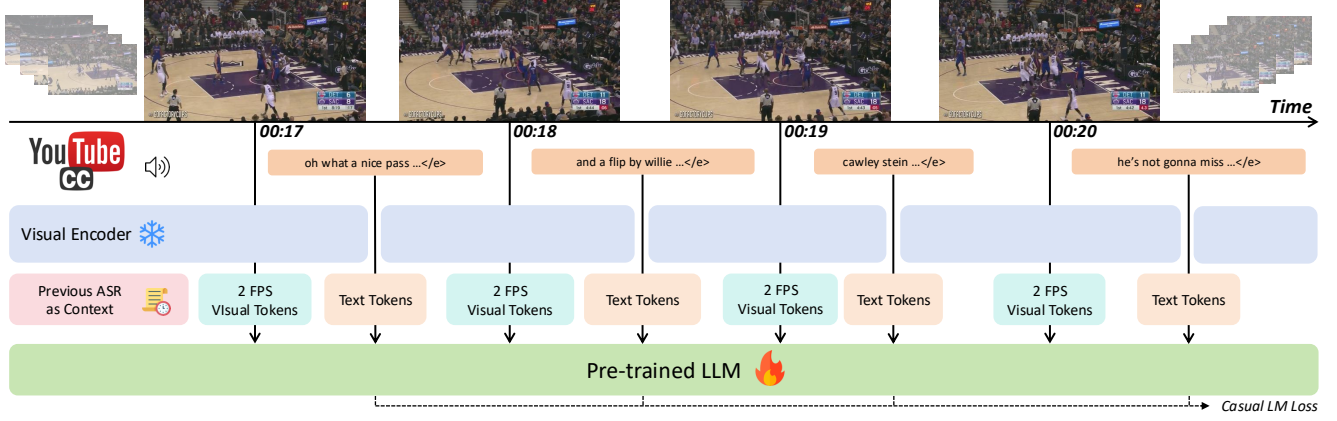


Figure 5. **Modeling Overview of LiveCC.** The model processes streaming video frames through a visual encoder to produce visual tokens while assigning ASR text from corresponding frame intervals as text tokens. The LLM autoregressively predicts text tokens within this densely interleaved token sequence. To mitigate learning ambiguity, additional context of preceding ASR text or video title is provided during pre-training. During SFT, the context part is only user query to match the real-world applications.

the LLM backbone. We adopt the *base* version of Qwen2-VL, which is pretrained extensively on image-text data but has limited exposure to video-text pairs. Following standard practice [54], the model is trained to autoregressively predict text tokens while treating visual tokens as non-predictive inputs, as illustrated in Figure 5. Unlike existing approaches that use either captioning [54] or image-text interleaving [50] style input, we propose to densely interleave ASR words with video frames along the temporal dimension. The training sequence is formatted as,

$$[\text{Con}] \langle F_{t:t+k} \rangle \langle W_{t:t+k} \rangle \langle F_{t+k:t+2k} \rangle \langle W_{t+k:t+2k} \rangle \\ \dots \langle F_{t+n*k:t+(n+1)*k} \rangle \langle W_{t+n*k:t+(n+1)*k} \rangle,$$

where $[\text{Con}]$ denotes context information of the video (e.g., prompt, previous ASR, video title), $\langle F \rangle$ denotes a frame, $\langle W \rangle$ denotes the words, t represents the time index and k represents the time intervals. By default, we use 2 FPS frame rate and $k = 1$ as the time interval. We incorporate video titles and preceding ASR text as contextual information to enhance text coherence, since ASR text may start from the middle of a sentence, or use informal, verbal language. A newline character concatenates the video title and previous ASR texts if the ASR texts are available.

Sequence Pre-processing. For pre-training, we utilize the original YouTube ASR transcripts, which employ fixed timestamps to segment speech into chunks of approximately 2 to 3 seconds. To approximate word-level alignment, we uniformly distribute each segment’s duration across its constituent words. This heuristic yields reasonably accurate word-level timestamps across the entire video. In contrast, during SFT, we leverage WhisperX, which provides precise word-level timestamps, as detailed in Section 3.1. To disambiguate the true end-of-sequence (EOS) from temporary pauses in streaming, we simply use the ellipsis token (“...”)

as a special EOS indicator appended to the per-frame text tokens. For silent frames without corresponding subtitles, we directly predict this ellipsis token.

Training Strategy. Our model training incorporates two stages including pre-training and SFT. For the pre-training, we solely train the model with dense interleaving sequences. The objective is to align frame features with the temporally synchronized ASR words, enabling the model to capture temporal correlations between frames and language. Next, to improve the ability of LiveCC models to solve a diverse set of downstream tasks, we jointly train the model with our Live-WhisperX-526K in streaming mode, general video and image datasets [105] for common caption or QA. To achieve this, we make the streaming training be compatible with the Qwen2-VL [80] conversation template. The details can be found in the supplementary material.

Inference. During inference, our LiveCC model processes input frames sequentially. To accelerate language decoding, we cache the Key-Value (KV) pairs of previous prompts, visual frames, and generated text. For long sequences, we discard visual tokens every 240 seconds—consistent with the maximum duration in SFT training—while retaining the text tokens to prefill the model again.

4. The LiveSports-3K Benchmark

4.1. LiveSports-3K Data Collection

As mentioned in Section 2, we present **LiveSports-3K**, a comprehensive benchmark designed for systematic evaluation of video understanding models’ capabilities. Unlike previous sports benchmarks like SoccerNet [61] and MatchTime [73], which focus on specific sports, our benchmark spans a broader range of common sports to ensure generalizability. To achieve this, we prompt GPT-4o-

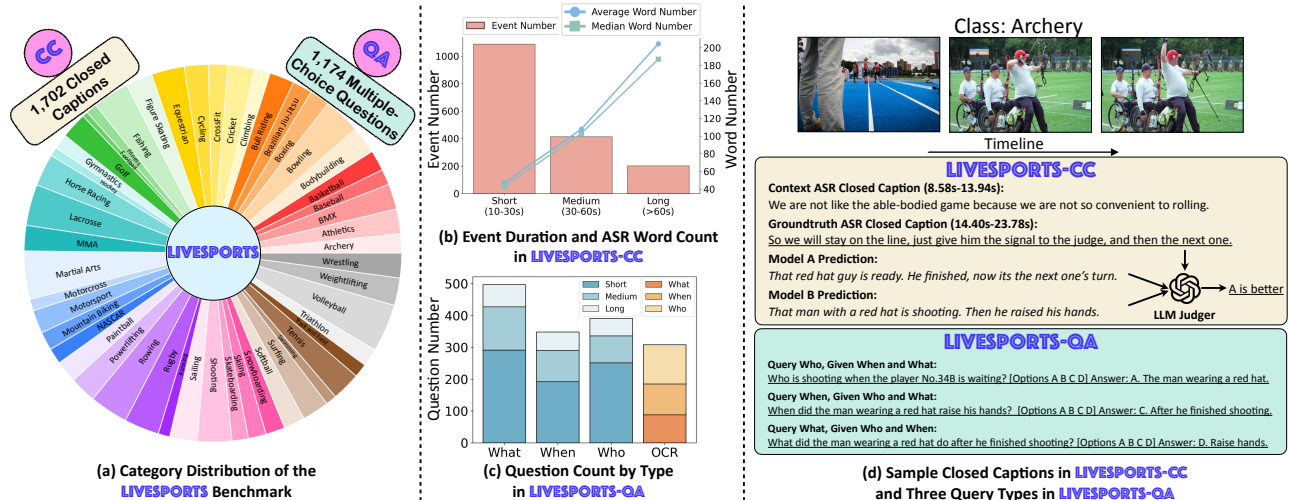


Figure 6. (a) Category Distribution of the LiveSports Benchmark: The benchmark includes 3k live CCs and MCQs, split into two tracks: CC and QA. (b) Event Duration and ASR Word Count in the CC track: For CC, event duration (left y-axis) and ASR word count (right y-axis) are analyzed, with durations categorized as short, medium, and long. (c) Question Count by Type in the QA track: Questions are grouped into three query types, with additional tracking of those requiring OCR for each type. (d) Sample CCs and Query Types.

mini [29] to select the ongoing sports videos and classify sports categories of our Live-WhisperX-526K dataset. We focused on the top 50 most frequent sports categories and randomly sampled 12 videos from each category, yielding a pool of 600 candidate videos covering popular sports. After collecting candidate videos, we used GPT-4o-mini to merge ASR transcriptions into semantically coherent events, recording each event’s start and end timestamps. Given that ASR transcriptions are not always visually relevant, we recruited English-proficient annotators to filter out irrelevant events according to three criteria: 1) The event must last more than 10 seconds; 2) The event clip must contain ongoing sports action; 3) Most ASR transcriptions within the event clip should be visually grounded. Events that failed to meet any of these criteria were discarded. We curated 416 videos across 49 sports categories (one category lack of qualified videos). The category distribution is shown in Figure 6(a). We further remove these videos from our training dataset for fair evaluation.

4.2. Crafting LiveSports-3K-CC/QA

LiveSports-3K-CC. Real-time commentary in sports videos encapsulates rich spatiotemporal semantics. For instance, the commentary on a football game often describes attackers, defenders, and their interactions in detail, making it an ideal source for streaming video understanding evaluations. Therefore, we developed this track to assess video comprehension by evaluating the alignment between model-generated and groundtruth CCs from ASR. Note that the data collection process has ensured that the ASR event transcriptions are visually grounded.

Thus, we directly leverage ASR transcriptions from qualified events as groundtruth. In summary, LiveSports-3K-CC consists of 1,702 events with high-quality live CCs. Figure 6(b) presents the distribution of events and word counts in three duration groups, showing a relatively balanced total word count (*i.e.*, video count \times word count per group) across groups. Additionally, Figure 6(d) illustrates a sample event, demonstrating the highly visual-grounded nature of the ASR-transcribed CCs retained through our filtering criteria.

For evaluation, we prompt the model with the video title and the preceding CCs of the event, then record the model’s predictions. Given the challenges of directly assessing discrepancies between the predicted and groundtruth CCs, we adopt a pairwise comparison approach inspired by Chatbot Arena [109]. Specifically, for each pair of predictions, we use GPT-4o as a judge to select the better prediction based on the ground-truth CCs. The selection criteria include both stylistic and semantic consistency. The winning rates of different models serve as the ranking metric.

LiveSports-3K-QA. LiveSports-3K-CC offers a valuable track for assessing video understanding comprehensively. However, it still lacks a precise criterion for analyzing model behavior, especially when errors occur. To address this, we decompose each event into three fundamental elements: (i) **When**: Captures the temporal context of the event. (ii) **What**: Defines the content or action taking place in the event. (iii) **Who**: Identifies the participants involved in the event. By structuring events around these elements, we enable targeted queries for each element based on the other two, allowing us to isolate specific areas of weak-

Video-ASR Sequence	LiveSports-3K-CC Win Rate↑	Video-MME	
		Overall↑	Short↑
Caption (5M)	14.0	61.1	69.4
Streaming (5M)	32.9	61.0	70.1
Caption+Streaming (5M×2)	35.1	60.5	69.0

Context	LiveSports-3K-CC Win Rate↑	Video-MME	
		Overall↑	Short↑
None	14.7	60.7	69.0
Title	24.8	59.7	67.9
Prev. ASR	32.0	61.1	69.7
Title & Prev. ASR	33.8	60.7	69.4
Title Prev. ASR	32.9	61.0	70.1

Data	LiveSports-3K-CC Win Rate↑	Video-MME	
		Overall↑	Short↑
1M	29.1	60.6	68.1
2.5M	30.8	60.9	69.1
5M	32.9	61.0	70.1
10M	36.0	58.0	67.6

(a) Training Paradigm during Pre-training.

(b) Context during Pre-training.

(c) Pre-training Scalability.

Table 1. Ablation Study in the Pre-training Stage. Blue highlights our default setting. Win Rate indicates the percentage of wins against GPT-4o-08-06-generated commentary, using ground-truth ASR as the reference. Models in these table are pre-trained on a maximum of 120 frames (60 seconds at 2 FPS), so we did not show results on medium- and long- length videos for Video-MME [23] to avoid misjudgment. (a) Both caption-style and streaming-style sequence significantly improve QA performance; however, only the streaming-style sequence yields notable gains in commentary generation. (b) Incorporating previous ASR enhances both commentary and QA. In contrast, simply adding video titles degrades QA, unless previous ASR is absent (*e.g.*, during 0–60s). (c) Commentary benefits from increased pre-training data, while QA performance declines beyond 5M examples—likely due to overtraining on the single-source (streaming ASR) data.

ness that require improvement. Figure 6(c) provides detailed examples of these three question types. Additionally, we recorded whether a question required OCR capabilities, allowing for an auxiliary evaluation of model performance on text recognition tasks. This process yielded 1,236 four-option MCQs across 414 videos, excluding two videos due to the difficulty of designing appropriate questions. Finally, we manually removed 62 questions that require speech recognition, leaving the remaining 1,174 MCQs as the final benchmark. This track includes a balanced distribution of the three query types, with OCR-reliant questions evenly distributed among them, as shown in Figure 6(b).

5. Experiments

5.1. Experiments Setup

Implementation Details. We initialize our model with the Qwen2-VL-7B-Base checkpoint [80], following most of the configurations provided in its HuggingFace release, with minimal modifications to improve efficiency. Specifically, during ablation studies of pre-training, we reduce the maximum number of frames from 768 to 120 and shorten the visual context length from 128K to 16K tokens. During formal pre-training and SFT, we increase the frame limit to 480 and extend the visual context length to 24K, while slightly lowering the minimum spatial resolution from $128 \times 28 \times 28$ to $100 \times 28 \times 28$. Pre-training ablation studies are conducted on the 30~60s Live-CC-1~10M dataset. The formal pre-training is in 30~240s Live-CC-5M. The SFT stages uses our Live-Whisper-526K and LLaVA-Video-178K [105] datasets (without the training set of ActivityNetQA [30], Next-QA [85], and PerceptionTest [66]). We implement the training engine using PyTorch [65] and Transformers [84]. The batch size for pre-training and SFT is 512 on 128 GPUs, with a learning rate of $2e-5$ for pre-training and $1e-5$ for SFT.

Evaluation Protocols and Metrics. For QA benchmarks, we evaluate our models on VideoMME [23], MVBench [45], OVOBench [46], and our newly introduced

LiveSports-3K-QA. For all models, we calculate the logits of multiple choices to select answers, due to the instruction following capability of streaming ASR pre-trained model has been lost. We observe this does not make difference with generation-based method [101] for SFT models, but the evaluation is much faster.

The evaluation on LiveSports-3K-CC is like a conditioned video captioning task, where the condition comprises the video title and previous ASR text. With this condition, the model’s task is to complete the ASR text based on the given video clip. Since most video LLMs lack real-time streaming capabilities, we evaluate them using a general video captioning approach, processing all video clip frames at once. In contrast, our model supports real-time inference, enabling us to generate captions on a frame-by-frame basis and then concatenate them into a complete response for evaluation. Due to the challenges of directly assessing open-ended text generation, we employ a pairwise competition approach, similar to Chatbot Arena [109]. We use GPT-4o [29] as the fixed competition opponent. In each competition (tested model vs. GPT-4o), we also use GPT-4o [29] acts as the judge, selecting the response that best aligns both stylistically and semantically with the ground truth ASR transcriptions. The evaluation metric is the win rate, which is defined as the proportion of times the judge favors our model over the baseline. The evaluation also involves latency comparison, which we discuss in the supplementary material.

5.2. Ablation Study

Pre-training Paradigm. We first investigate the impact of different pre-training paradigms on model performance using the following baselines: (1) Caption, where all ASR text is concatenated and appended after the visual input frames; (2) Streaming, where the model is trained on sequential frame inputs sampled at 2 FPS, predicting ASR text incrementally after receiving every 2 frames; and (3) Caption+Streaming, where each training sample contributes to both captioning and streaming objectives. As shown in

Pre-training	SFT	LiveSports-3K							VideoMME														
		CC	QA	OCR	Who	When	What	All	Duration			Perception			Recognition		Reasoning				OCR	Count	IS
Qwen2-VL-7B-Base	LV178K	16.7	67.0	66.1	70.6	57.6	71.0	62.7	74.7	62.4	51.1	74.5	61.1	73.4	64.5	70.1	49.7	80.4	49.8	57.0	76.3	45.1	76.2
	LV178K+Live526K	33.7	67.1	66.8	69.8	57.0	72.3	63.6	74.4	63.1	53.2	74.5	57.4	75.2	66.5	70.1	49.7	76.8	54.4	57.5	72.7	44.4	78.9

(a) Ablation study in the SFT data.

Pre-training	SFT	LiveSports-3K							VideoMME														
		CC	QA	OCR	Who	When	What	All	Duration			Perception			Recognition		Reasoning				OCR	Count	IS
Qwen2-VL-7B-Base	-	16.3	64.0	64.8	65.2	57.9	67.3	63.4	73.2	63.2	53.9	72.7	63.0	76.1	63.9	67.2	44.6	78.6	57.5	61.5	72.7	39.6	80.2
LiveCC-7B-Base	-	43.2	57.9	61.4	59.4	50.7	61.9	61.4	68.1	58.9	57.3	65.5	63.0	64.9	60.7	61.0	50.3	80.4	56.1	61.5	61.2	42.9	82.4
Qwen2-VL-7B-Base	LV178K+Live526K	33.7	67.1	66.8	69.8	57.0	72.3	63.6	74.4	63.1	53.2	74.5	57.4	75.2	66.5	70.1	49.7	76.8	54.4	57.5	72.7	44.4	78.9
LiveCC-7B-Base	LV178K+Live526K	41.5	66.8	66.4	71.4	56.1	70.8	64.1	74.8	63.9	53.7	74.5	64.8	74.3	66.1	68.6	50.3	76.8	52.3	59.5	77.0	46.3	79.9

(b) Ablation study in the SFT model initialization.

Table 2. Ablation studies in the SFT stage. LV178K denotes the datasets used in LLaVA-Video-178K [105]. Live526K refers to our proposed Live-WhisperX-526K. LiveSports-3K CC denotes the win rate against commentary generated by LLaVA-Video-72B. LiveSports-3K QA is the overall accuracy includes OCR, Who, When, What questions. Te, Sp, At, Ac, Ob, IS denotes temporal, spatial, attribute, action, object, information synopsis, respectively.

Model (7B/8B)	VideoMME		MVBench	OVOBench			
	w/o sub	w sub	Avg.	Avg.	RTVP	BT	FAR
LongVA-7B [103]	52.6	54.3	-	-	-	-	-
InternVL2-8B [17]	54.0	56.9	66.4	50.2	60.4	43.4	46.6
LLaVA-OV-7B [41]	58.2	61.5	56.7	52.7	64.0	43.7	50.5
Oryx-7B [56]	58.3	62.6	63.9	-	-	-	-
mPLUG-Owl3-7B [94]	59.3	68.1	59.5	-	-	-	-
LongVU-7B [75]	60.6	-	66.9	46.7	57.6	35.0	47.5
MiniCPM-v2.6 [92]	60.9	63.6	-	-	-	-	-
Qwen2-VL-7B-Instruct [80]	63.3	69.0	67.0	50.4	56.0	46.5	48.7
LLaVA-Video-7B [106]	63.3	69.7	58.6	52.9	63.5	40.4	54.8
LiveCC-7B-Instruct	64.1	70.3	62.8	59.8	59.1	68.9	51.5

Table 3. Comparison of QA accuracy (%) across VideoMME [23], MVBench [45], OVOBench [46]. We only show results before the CVPR 2025 submission period (Nov, 2024).

Table 1a, both caption and streaming pre-training achieve strong general video QA performance (exceeding 60) on the Video-MME benchmark [23], outperforming many existing SFT models. Notably, the streaming-based pre-training yields significantly better results on the commentary task compared to caption-based pre-training, highlighting the effectiveness of our proposed paradigm.

Context Input for Pre-training. In Table 1b, we observe that providing contextual information, particularly the previous ASR text, significantly improves commentary generation. This improvement stems from the fact that a 60-second segment can break the continuity of ASR, making it difficult to interpret the current segment without prior context. While incorporating the video title as additional context offers benefits on commentary, it slightly degrades performance on VideoMME. We attribute this to potential information leakage, which may make training easier. To handle the cases where no previous ASR is available (e.g., clips at the beginning of a video), we adopt a hybrid strategy: “Title || Prev. ASR”, which includes the video title only when previous ASR is unavailable. This approach strikes the best balance between enhancing commentary generation

Size	Model	Live?	LiveSports-3K ↑						
			CC	Overall	OCR	Who	When	What	
72B	GPT-4o-08-06 [29]	✗	✗	72.2	74.0	75.8	63.4	75.4	
	Gemini-1.5-Pro [3]	✗	52.8	61.8	61.7	59.9	51.6	70.7	
	Qwen2-VL-72B-Instruct [80]	✗	17.0	70.8	67.8	74.6	61.2	74.6	
	VideoLLaMA-2-72B [18]	✗	24.8	62.4	55.7	63.6	54.3	67.3	
	LLaVA-OV-72B [41]	✗	29.2	68.7	61.7	71.1	61.5	71.8	
7B	Qwen2.5-VL-72B-Instruct [6]	✗	30.4	73.7	70.1	75.7	69.3	75.3	
	LLaVA-Video-72B [106]	✗	35.0	71.1	65.1	74.1	64.8	73.3	
	Qwen2-VL-7B-Instruct [80]	✗	9.3	65.8	65.8	67.9	58.8	69.2	
	Qwen2.5-VL-7B-Instruct [6]	✗	17.3	67.0	64.8	70.3	60.6	69.0	
	InternLM-XC2.5-7B [102]	✗	17.3	59.3	56.7	60.7	54.9	61.3	
	Qwen2.5-Omni-7B [87] (Thinker)	✗	17.6	66.8	66.1	70.0	60.0	69.2	
	LLaVA-Video-7B [106]	✗	27.1	66.4	64.1	72.7	56.4	68.6	
	LLaVA-OV-7B [41]	✗	27.7	63.4	60.7	67.4	53.7	67.1	
	Qwen2-VL-7B-LiveCCInstruct	✓	33.7	67.1	66.8	69.8	57.0	72.3	
	LiveCC-7B-Instruct	✓	41.5	66.8	66.4	71.4	56.1	70.8	
	LiveCC-7B-Base	✓	43.2	57.9	61.4	59.4	50.7	61.9	

Table 4. Win rate on the LiveSports-3K-CC track and QA accuracy on the LiveSports-3K-QA track. GPT-4o-08-06 is used as the **Baseline** for the commentary win rate comparison due to its strong performance. For a fair comparison, all Qwen models [6, 80, 87] and our models are evaluated on a maximum of 480 frames. The Qwen models did not perform as expected, as they tend to simply caption the video rather than follow the preceding ASR context to continue the video commentary.

and maintaining general video QA performance.

Pre-training Scalability. Table 1c presents the results of scaling up the pre-training data. We observe that commentary performance consistently improves with larger data size. However, QA performance begins to decline beyond the 5M scale, likely due to the use of single-source (streaming commentary) data during pre-training. Since our primary goal is to demonstrate the effectiveness of the streaming-based pre-training, we defer the use of multi-source data to the SFT stage.

5.3. Overall Results

In Table 3 and Table 4, we compare the performance of various models on general QA, streaming QA and



Figure 7. **Comparison of pre-trained and instruction tuning enhanced model's predictions on the same video.** This example is sourced from Video-MME [23], with the YouTube ID whksDmTR9YE featuring animal fights.

our LiveSports-3K benchmarks, including advanced proprietary models, SOTA open-source 72B models, and SOTA open-source 7B models. Despite being initialized from Qwen2-VL-7B-Base, our LiveCC-7B-Instruct outperforms Qwen2-VL-7B-Instruct on the general QA, i.e., VideoMME (64.1 vs. 63.3) and the streaming benchmark, i.e., OVOBench (59.8 vs. 50.4). This demonstrates the strong generalization capabilities of our dataset and training method. In our proposed LiveSports-3K benchmark, we observe that our three models achieve significant advantages in commentary while maintain competitive on QA, which demonstrates the effectiveness of our method.

5.4. Streaming Commentary Capabilities

Figure 7 shows that the pre-trained model can already demonstrate impressive real-time video commentary capabilities. With SFT, the model further improves formatting (e.g., punctuation, case) and coherence. More examples are provided in the supplementary material.

6. Conclusion

In this paper, we investigated the large-scale training of video LLMs using ASR transcripts. We proposed a novel streaming training approach that densely interleaves fine-grained ASR words with their corresponding video frames based on timestamps. Our methodology involved the collection of two datasets: Live-CC-5M for pre-training and Live-WhisperX-526K for instruction tuning. We then developed our streaming pre-training approach, introducing a series of innovative training and inference strategies. Additionally, we designed LiveSports-3K, with two evaluation tracks, LiveSports-3K-CC and LiveSports-3K-QA, which are specifically tailored to assess the model's streaming capabilities. Our extensive experiments demonstrate that our model can perform low-latency commentary for streaming videos and general question answering for holistic video un-

derstanding in state-of-the-art performance simultaneously. In future work, we seek methods to train multimodal omni models in streaming.

Acknowledgments

This research is supported by the National Research Foundation, Singapore, under its AI Singapore Programme (AISG Award No: AISG3-RP-2022-030). Joya Chen proposed the idea, designed and implemented the data production pipeline and the model training/inference codebase. Ziyun Zeng built the benchmark and designed and implemented the free-form commentary evaluation. Yiqi Lin made significant contributions to baseline model analysis, data collection, and paper writing. We jointly trained and evaluated the models and discussed the results together. We acknowledge the resource support from TikTok AIIC and thank Wei Li and Zejun Ma for their valuable guidance. We are also grateful for the insightful discussions with Rui Qian and Yu Li, and we thank Kevin Qinghong Lin, Zhaoyang Lv, Yichi Zhang, and Huiyu Wang for their constructive comments.

References

- [1] Huda AlAmri, Vincent Cartillier, Abhishek Das, Jue Wang, Anoop Cherian, Irfan Essa, Dhruv Batra, Tim K. Marks, Chiori Hori, Peter Anderson, Stefan Lee, and Devi Parikh. Audio visual scene-aware dialog. In *CVPR*, pages 7558–7567, 2019. 3
- [2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L. Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022. 2
- [3] Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Slav Petrov, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy P. Lillicrap, Angeliki Lazaridou, Orhan Firat, James Molloy, Michael Isard, Paul Ronald Barham, Tom Hennigan, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, Ryan Doherty, Eli Collins, Clemens Meyer, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, George Tucker, Enrique Piqueras, Maxim Krikun, Iain Barr, Nikolay Savinov, Ivo Danihelka, Becca Roelofs, Anaïs White, Anders Andreassen, Tamara von Glehn, Lakshman Yagati, Mehran Kazemi, Lucas Gonzalez, Misha Khalman, Jakub Sygnowski, and et al. Gemini: A family of highly capable multimodal models. *arXiv:2312.11805*, 2023. 1, 2, 8, 15

- [4] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. Qwen technical report. *arXiv:2309.16609*, 2023. 1
- [5] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv:2308.12966*, 2023. 2
- [6] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv:2412.15115*, 2024. 2, 8, 15
- [7] Max Bain, Jaesung Huh, Tengda Han, and Andrew Zisserman. Whisperx: Time-accurate speech transcription of long-form audio. *arXiv preprint arXiv:2303.00747*, 2023. 3, 4
- [8] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *NeurIPS*, pages 1877–1901, 2020. 2
- [9] Shyamal Buch, Victor Escorcia, Chuanqi Shen, Bernard Ghanem, and Juan Carlos Niebles. Sst: Single-stream temporal action proposals. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2911–2920, 2017. 3
- [10] João Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *CVPR*, pages 4724–4733, 2017. 3
- [11] João Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A short note about kinetics-600. *Arxiv*, 1808.01340, 2018. 3
- [12] Jun Chen, Deyao Zhu, Xiaoqian Shen, Xiang Li, Zechun Liu, Pengchuan Zhang, Raghuraman Krishnamoorthi, Vikas Chandra, Yunyang Xiong, and Mohamed Elhoseiny. Minigt-v2: large language model as a unified interface for vision-language multi-task learning. *arXiv:2310.09478*, 2023. 2
- [13] Joya Chen, Zhaoyang Lv, Shiwei Wu, Kevin Qinghong Lin, Chenan Song, Difei Gao, Jia-Wei Liu, Ziteng Gao, Dongxing Mao, and Mike Zheng Shou. Videollm-online: Online video large language model for streaming video. In *CVPR*, 2024. 2, 3
- [14] 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. *arXiv preprint arXiv:2406.04325*, 2024. 2
- [15] Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang-wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, et al. Panda-70m: Captioning 70m videos with multiple cross-modality teachers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13320–13331, 2024. 2
- [16] Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, Lixin Gu, Xuehui Wang, Qingyun Li, Yimin Ren, Zixuan Chen, Jiapeng Luo, Jiahao Wang, Tan Jiang, Bo Wang, Conghui He, Botian Shi, Xingcheng Zhang, Han Lv, Yi Wang, Wenqi Shao, Pei Chu, Zhongying Tu, Tong He, Zhiyong Wu, Huipeng Deng, Jiaye Ge, Kai Chen, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhao Wang. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv:2412.05271*, 2024. 2
- [17] Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *arXiv preprint arXiv:2404.16821*, 2024. 8
- [18] Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, et al. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. *arXiv preprint arXiv:2406.07476*, 2024. 8
- [19] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *ACL*, 2020. 3
- [20] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven C.H.Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *arXiv:2305.06500*, 2023. 2
- [21] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 4
- [22] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurélien Rodriguez, Austen

- Gregerson, Ava Spataru, Baptiste Rozière, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Grégoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, and et al. The llama 3 herd of models. *arXiv:2407.21783*, 2024. 1
- [23] Chaoyou Fu, Yuhan Dai, Yondong 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. *arXiv preprint arXiv:2405.21075*, 2024. 2, 7, 8, 9
- [24] Chaoyou Fu, Haojia Lin, Zuwei Long, Yunhang Shen, Meng Zhao, Yifan Zhang, Shaoqi Dong, Xiong Wang, Di Yin, Long Ma, et al. Vita: Towards open-source interactive omni multimodal llm. *arXiv:2408.05211*, 2024. 2, 3
- [25] Chaoyou Fu, Haojia Lin, Xiong Wang, Yi-Fan Zhang, Yunhang Shen, Xiaoyu Liu, Yangze Li, Zuwei Long, Heting Gao, Ke Li, et al. Vita-1.5: Towards gpt-4o level real-time vision and speech interaction. *arXiv:2501.01957*, 2025. 2, 3
- [26] Roeland De Geest, Efstratios Gavves, Amir Ghodrati, Zhenyang Li, Cees Snoek, and Tinne Tuytelaars. Online action detection. In *ECCV*, pages 269–284, 2016. 3
- [27] Gemini Team Google. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv:2403.05530*, 2024. 2
- [28] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fründ, Peter Yianilos, Moritz Mueller-Freitag, Florian Hoppe, Christian Thureau, Ingo Bax, and Roland Memisevic. The “something something” video database for learning and evaluating visual common sense. In *ICCV*, pages 5843–5851, 2017. 3
- [29] GPT-4o. Hello gpt-4o, 2024. 2, 4, 6, 7, 8, 15
- [30] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, pages 961–970, 2015. 3, 7
- [31] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models. *arXiv:2203.15556*, 2022. 2
- [32] Gabriel Huang, Bo Pang, Zhenhai Zhu, Clara Rivera, and Radu Soricut. Multimodal pretraining for dense video captioning. In *IJCNLP-AACL*, pages 470–490, 2020. 2
- [33] Zhenpeng Huang, Xinhao Li, Jiaqi Li, Jing Wang, Xianguyu Zeng, Cheng Liang, Tao Wu, Xi Chen, Liang Li, and Limin Wang. Online video understanding: A comprehensive benchmark and memory-augmented method. *arXiv:2501.00584*, 2025. 3
- [34] Haroon Idrees, Amir R. Zamir, Yu-Gang Jiang, Alex Gorbunov, Ivan Laptev, Rahul Sukthankar, and Mubarak Shah. The THUMOS challenge on action recognition for videos “in the wild”. *Comput. Vis. Image Underst.*, 155:1–23, 2017. 3
- [35] Hyolim Kang, Kyungmin Kim, Yumin Ko, and Seon Joo Kim. Cag-qil: Context-aware actionness grouping via q imitation learning for online temporal action localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13729–13738, 2021. 3
- [36] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv:2001.08361*, 2020. 2
- [37] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *ICCV*, pages 706–715, 2017. 2
- [38] Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. *arXiv:2308.00692*, 2023. 2
- [39] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L. Berg. TVQA: localized, compositional video question answering. In *EMNLP*, pages 1369–1379, 2018. 3
- [40] Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Jingkang Yang, and Ziwei Liu. Otter: A multi-modal model with in-context instruction tuning. *arXiv:2305.03726*, 2023. 2
- [41] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. 8, 15
- [42] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv:2408.03326*, 2024. 2
- [43] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 2
- [44] KunChang Li, Yanan He, Yi Wang, Yizhuo Li, Wenhai Wang, Ping Luo, Yali Wang, Limin Wang, and

- Yu Qiao. Videochat: Chat-centric video understanding. *arXiv:2305.06355*, 2023. 2
- [45] 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 *CVPR*, pages 22195–22206, 2024. 2, 7, 8
- [46] Yifei Li, Junbo Niu, Ziyang Miao, Chunjiang Ge, Yuanhang Zhou, Qihao He, Xiaoyi Dong, Haodong Duan, Shuangrui Ding, Rui Qian, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, and Jiaqi Wang. Ovo-bench: How far is your video-llms from real-world online video understanding? *arXiv:2501.05510*, 2025. 2, 3, 7, 8
- [47] Junhua Liao, Haihan Duan, Kanghui Feng, Wanbing Zhao, Yanbing Yang, and Liangyin Chen. A light weight model for active speaker detection. In *CVPR*, pages 22932–22941, 2023. 2, 4
- [48] Bin Lin, Bin Zhu, Yang Ye, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. *arXiv:2311.10122*, 2023. 2
- [49] Junming Lin, Zheng Fang, Chi Chen, Zihao Wan, Fuwen Luo, Peng Li, Yang Liu, and Maosong Sun. Streaming-bench: Assessing the gap for mllms to achieve streaming video understanding. *arXiv preprint arXiv:2411.03628*, 2024. 3
- [50] Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shoeybi, and Song Han. VILA: on pre-training for visual language models. In *CVPR*, pages 26679–26689, 2024. 2, 5
- [51] Kevin Qinghong Lin, Alex Jinpeng Wang, Mattia Soldan, Michael Wray, Rui Yan, Eric Zhongcong Xu, Difei Gao, Rongcheng Tu, Wenzhe Zhao, Weijie Kong, Chengfei Cai, Hongfa Wang, Dima Damen, Bernard Ghanem, Wei Liu, and Mike Zheng Shou. Egocentric video-language pretraining. *arXiv:2206.01670*, 2022. 3
- [52] Kevin Qinghong Lin, Pengchuan Zhang, Joya Chen, Shraman Pramanick, Difei Gao, Alex Jinpeng Wang, Rui Yan, and Mike Zheng Shou. Univt: Towards unified video-language temporal grounding. In *ICCV*, pages 2782–2792, 2023. 2
- [53] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv:2310.03744*, 2023. 2
- [54] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *NeurIPS*, 2023. 2, 5
- [55] Jihao Liu, Zhiding Yu, Shiyi Lan, Shihao Wang, Rongyao Fang, Jan Kautz, Hongsheng Li, and Jose M. Alvarez. Streamchat: Chatting with streaming video. *arXiv:2412.08646*, 2024. 2, 3
- [56] 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. 8
- [57] 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)*, pages 12585–12602, Bangkok, Thailand, 2024. Association for Computational Linguistics. 2
- [58] Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. In *NeurIPS*, pages 46212–46244, 2023. 3
- [59] Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter, Dhruvi Shah, Xianzhi Du, Futang Peng, Floris Weers, et al. Mm1: Methods, analysis & insights from multimodal llm pre-training. *arXiv preprint arXiv:2403.09611*, 2024. 2
- [60] 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*, pages 2630–2640, 2019. 2, 3
- [61] Hassan Mkhallati, Anthony Cioppa, Silvio Giancola, Bernard Ghanem, and Marc Van Droogenbroeck. Soccernet-caption: Dense video captioning for soccer broadcasts commentaries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5074–5085, 2023. 5
- [62] OpenAI. Introducing chatgpt. <https://openai.com/blog/chatgpt/>, 2023. 1
- [63] OpenAI. GPT-4 technical report. *arXiv:2303.08774*, 2023. 1
- [64] OpenAI. Gpt-4v(ision) system card. https://cdn.openai.com/papers/GPTV_System_Card.pdf, 2023. 2
- [65] Adam Paszke, Sam Gross, and Francisco et al Massa. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, pages 8026–8037, 2019. 7
- [66] Viorica Patraucean, Lucas Smaira, Ankush Gupta, Adria Recasens, Larisa Markeeva, Dylan Banarse, Skanda Kopula, Mateusz Malinowski, Yi Yang, Carl Doersch, et al. Perception test: A diagnostic benchmark for multimodal video models. *Advances in Neural Information Processing Systems*, 36, 2024. 7
- [67] Rui Qian, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Shuangrui Ding, Dahua Lin, and Jiaqi Wang. Streaming long video understanding with large language models. In *NeurIPS*, 2024. 2, 3
- [68] Rui Qian, Shuangrui Ding, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Dahua Lin, and Jiaqi Wang. Dispider: Enabling video llms with active real-time interaction via disentangled perception, decision, and reaction. *arXiv:2501.03218*, 2025. 2, 3
- [69] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018. 2
- [70] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. 2
- [71] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen

- Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021. 2
- [72] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *ICML*, pages 28492–28518, 2023. 3, 4
- [73] Jiayuan Rao, Haoning Wu, Chang Liu, Yanfeng Wang, and Weidi Xie. Matchtime: Towards automatic soccer game commentary generation. *arXiv preprint arXiv:2406.18530*, 2024. 5
- [74] Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. Timechat: A time-sensitive multimodal large language model for long video understanding. In *CVPR*, pages 14313–14323, 2024. 2
- [75] Xiaoqian Shen, Yongyang Xiong, Changsheng Zhao, Lemeng Wu, Jun Chen, Chenchen Zhu, Zechun Liu, Fanyi Xiao, Balakrishnan Varadarajan, Florian Bordes, Zhuang Liu, Hu Xu, Hyunwoo J. Kim, Bilge Soran, Raghuraman Krishnamoorthi, Mohamed Elhoseiny, and Vikas Chandra. Longvu: Spatiotemporal adaptive compression for long video-language understanding. *arXiv:2410.17434*, 2024. 2, 8
- [76] Gurkirt Singh, Suman Saha, Michael Sapienza, Philip HS Torr, and Fabio Cuzzolin. Online real-time multiple spatiotemporal action localisation and prediction. In *Proceedings of the IEEE international conference on computer vision*, pages 3637–3646, 2017. 3
- [77] Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Xun Guo, Tian Ye, Yan Lu, Jenq-Neng Hwang, et al. Moviechat: From dense token to sparse memory for long video understanding. *arXiv:2307.16449*, 2023. 2
- [78] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv:2302.13971*, 2023. 1
- [79] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shrutu Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models. *arXiv:2307.09288*, 2023. 1
- [80] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv:2409.12191*, 2024. 2, 4, 5, 7, 8, 15
- [81] Yi Wang, Yanan He, Yizhuo Li, Kunchang Li, Jiahuo 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:2307.06942*, 2023. 2
- [82] Yueqian Wang, Xiaojun Meng, Yuxuan Wang, Jianxin Liang, Jiansheng Wei, Huishuai Zhang, and Dongyan Zhao. Videollm knows when to speak: Enhancing time-sensitive video comprehension with video-text duet interaction format, 2024. 2, 3
- [83] Yuxuan Wang, Cihang Xie, Yang Liu, and Zilong Zheng. Videollamb: Long-context video understanding with recurrent memory bridges. *arXiv:2409.01071*, 2024. 2, 3
- [84] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, 2020. Association for Computational Linguistics. 7
- [85] 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, 7
- [86] Haomiao Xiong, Zongxin Yang, Jiazuo Yu, Yunzhi Zhuge, Lu Zhang, Jiawen Zhu, and Huchuan Lu. Streaming video understanding and multi-round interaction with memory-enhanced knowledge. In *ICLR*, 2025. 2, 3
- [87] Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, et al. Qwen2.5-omni technical report. *arXiv preprint arXiv:2503.20215*, 2025. 8, 15
- [88] Hongwei Xue, Tiankai Hang, Yanhong Zeng, Yuchong Sun, Bei Liu, Huan Yang, Jianlong Fu, and Baining Guo. Advancing high-resolution video-language representation with large-scale video transcriptions. In *CVPR*, pages 5026–5035, 2022. 2, 3
- [89] Antoine Yang, Arsha Nagrani, Ivan Laptev, Josef Sivic, and Cordelia Schmid. Vidchapters-7m: Video chapters at scale. In *NeurIPS*, 2023. 3
- [90] Antoine Yang, Arsha Nagrani, Paul Hongsuck Seo, Antoine Miech, Jordi Pont-Tuset, Ivan Laptev, Josef Sivic, and Cordelia Schmid. Vid2seq: Large-scale pretraining of a visual language model for dense video captioning. In *CVPR*, pages 10714–10726, 2023. 2, 3

- [91] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024. 1, 4
- [92] Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint arXiv:2408.01800*, 2024. 8
- [93] Zhewei Yao, Xiaoxia Wu, Conglong Li, Minjia Zhang, Heyang Qi, Olatunji Ruwase, Ammar Ahmad Awan, Samyam Rajbhandari, and Yuxiong He. DeepSpeed-visualchat: Multi-round multi-image interleaved chat via multi-modal causal attention. *arXiv:2309.14327*, 2023. 2
- [94] 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. 8
- [95] Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang. Ferret: Refer and ground anything anywhere at any granularity. *arXiv:2310.07704*, 2023. 2
- [96] Rowan Zellers, Jiasen Lu, Ximing Lu, Youngjae Yu, Yanpeng Zhao, Mohammadreza Salehi, Aditya Kusupati, Jack Hessel, Ali Farhadi, and Yejin Choi. MERLOT RESERVE: neural script knowledge through vision and language and sound. In *CVPR*, pages 16354–16366, 2022. 2, 3
- [97] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *ICCV*, pages 11941–11952, 2023. 2
- [98] Boqiang Zhang, Kehan Li, Zesen Cheng, Zhiqiang Hu, Yuqian Yuan, Guanzheng Chen, Sicong Leng, Yuming Jiang, Hang Zhang, Xin Li, Peng Jin, Wenqi Zhang, Fan Wang, Lidong Bing, and Deli Zhao. Videollama 3: Frontier multimodal foundation models for image and video understanding. *arXiv:2501.13106*. 2, 3
- [99] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv:2306.02858*, 2023. 2
- [100] Haoji Zhang, Yiqin Wang, Yansong Tang, Yong Liu, Jiashi Feng, Jifeng Dai, and Xiaojie Jin. Flash-vstream: Memory-based real-time understanding for long video streams. *arXiv:2406.08085*, 2024. 2
- [101] Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, and Ziwei Liu. Lmms-eval: Reality check on the evaluation of large multimodal models. *arXiv:2407.12772*, 2024. 7
- [102] Pan Zhang, Xiaoyi Dong, Yuhang Zang, Yuhang Cao, Rui Qian, Lin Chen, Qipeng Guo, Haodong Duan, Bin Wang, Linke Ouyang, et al. Internlm-xcomposer-2.5: A versatile large vision language model supporting long-contextual input and output. *arXiv preprint arXiv:2407.03320*, 2024. 8
- [103] Peiyuan Zhang, Kaichen Zhang, Bo Li, Guangtao Zeng, Jingkang Yang, Yuanhan Zhang, Ziyue Wang, Haoran Tan, Chunyuan Li, and Ziwei Liu. Long context transfer from language to vision. *arXiv preprint arXiv:2406.16852*, 2024. 8
- [104] Shilong Zhang, Peize Sun, Shoufa Chen, Min Xiao, Wenqi Shao, Wenwei Zhang, Kai Chen, and Ping Luo. Gpt4roi: Instruction tuning large language model on region-of-interest. *arXiv:2307.03601*, 2023. 2
- [105] Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Video instruction tuning with synthetic data. *arXiv:2410.02713*, 2024. 2, 3, 5, 7, 8
- [106] Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Video instruction tuning with synthetic data. *arXiv preprint arXiv:2410.02713*, 2024. 2, 8, 15, 16
- [107] Yue Zhao and Philipp Krähenbühl. Real-time online video detection with temporal smoothing transformers. In *European Conference on Computer Vision*, pages 485–502. Springer, 2022. 3
- [108] Yucheng Zhao, Chong Luo, Chuanxin Tang, Dongdong Chen, Noel Codella, and Zheng-Jun Zha. Streaming video model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14602–14612, 2023. 3
- [109] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhaghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In *NeurIPS*, 2023. 2, 6, 7
- [110] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, 2018. 2
- [111] Xingyi Zhou, Anurag Arnab, Shyamal Buch, Shen Yan, Austin Myers, Xuehan Xiong, Arsha Nagrani, and Cordelia Schmid. Streaming dense video captioning. In *CVPR*, pages 18243–18252, 2024. 2, 3
- [112] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv:2304.10592*, 2023. 2

LiveCC: Learning Video LLM with Streaming Speech Transcription at Scale

Supplementary Material

7. Demo

This section showcases four demo videos to demonstrate the capability of our LiveCC-7B-Instruct to provide real-time commentary in real-world videos across different domains, including sports (football), science (astronomy), news (weather forecast), and instructional (computer repair) videos. As illustrated in Figure 10-13, in the first demo, our LiveCC-7B-Instruct model correctly recognizes all exact penalty timings, highlighting its strong temporal perception abilities. By leveraging the extensive world knowledge gained from watching millions of YouTube videos, our model accurately reports the name of the related player. The second demo showcases the model’s ability to comment beyond sports by precisely presenting astronomy knowledge and demonstrating good OCR capability to read large numbers. The third demo further reveals its fine-grained temporal understanding capability, as evidenced by its real-time commentary on subtle changes in weather maps. The final demo demonstrates that our model is also capable of generating a tutorial to guide users, revealing its potential to serve as a real-time assistant.

8. Implementation Details

8.1. Prompt Template

In this section, we detail the prompt designs used during the pre-training, instruction tuning, and inference stages. As shown in Figure 8(a) and (b), the video title and previously transcribed ASR text are provided as contextual information during pre-training but are omitted during SFT. For the first round of vision token extraction, we use a 3-second video clip, followed by 1-second clips in subsequent rounds. Given a frame rate of 2 FPS, this corresponds to 6 and 2 frames, respectively. For QA-style data used exclusively in SFT, illustrated in Figure 8(c), we adopt the input format of Qwen2-VL-Instruct [80], which is also used during evaluation. However, for real-time commentary evaluation, we follow the format in Figure 8(d), where the video title and previous ASR transcripts are included to ensure consistency with other non-streaming baselines.

8.2. Win Rate Computation on LiveSports-3K

In this section, we present the detailed process for computing the win rate on LiveSports-3K-CC. To start, we categorize the models into two groups based on their inference schemes: **i) Clip-wise caption models**, including GPT-4o [29], Gemini-1.5-Pro [3], LLaVA-OV-7/72B [41], LLaVA-Video-7/72B [106], Qwen2-

VL-7/72B-Instruct [80], Qwen2.5-VL-7/72B-Instruct [6] and Qwen2.5-Omni-7B [87]. **ii) Frame-wise streaming model**, *i.e.*, our LiveCC-7B-Base, Qwen2-VL-7B-LiveCCInstruct, LiveCC-7B-Instruct.

For clip-wise caption models, we directly input the overall event clips, perform a **one-time** generation, and use the generated response as the commentary. To ensure stylistic consistency and fair evaluation, the same prompt context as that shown in Figure 8(d) is applied across all models. We use the video commentary from GPT-4o-08-06 [29] serves as the baseline for comparison with other models. For our models, we adopt **streaming** inference shown in Figure 8(d), where commentary is generated frame by frame. The generated tokens are then concatenated to form the complete commentary, which is evaluated for quality.

For evaluation, we also prompt GPT-4o-08-06 [29] to assess whether a given commentary surpasses that of GPT-4o-08-06. The evaluation is based on two key criteria: **(i) Semantic Alignment**, *i.e.*, consider which text conveys the same meaning, details, and key points as the groundtruth ASR transcript, with minimal deviation. **(ii) Stylistic Consistency**, *i.e.*, assesses which text maintains a tone, word choice, and structure similar to the ground-truth transcript. The overall prompt is written as:

```
You are an expert in video commentary. Your task
is to review two commentaries (Commentary A and
Commentary B), and select the one that better
aligns with the human commentary. You should
consider the criteria:
1. Semantic Alignment: The commentary should
convey the same meaning, details, and key points
as the human commentary.
If the above criteria is not enough to judge,
then consider:
2. Stylistic Consistency: The commentary should
maintain a tone, word choice, and structure
similar to the human commentary.
---Commentary A---
{a_pred}
-----
---Commentary B---
{b_pred}
-----
---Human Commentary---
{gt_asr}
-----
Your response should be "Commentary A is better
aligned with the human commentary" or "Commentary
B is better aligned with the human commentary".
```

The final win rate is calculated as the proportion of instances where GPT-4o [29] selects the model’s response over the baseline. To mitigate positional bias in GPT’s responses, each prompt is evaluated *twice* with the positions

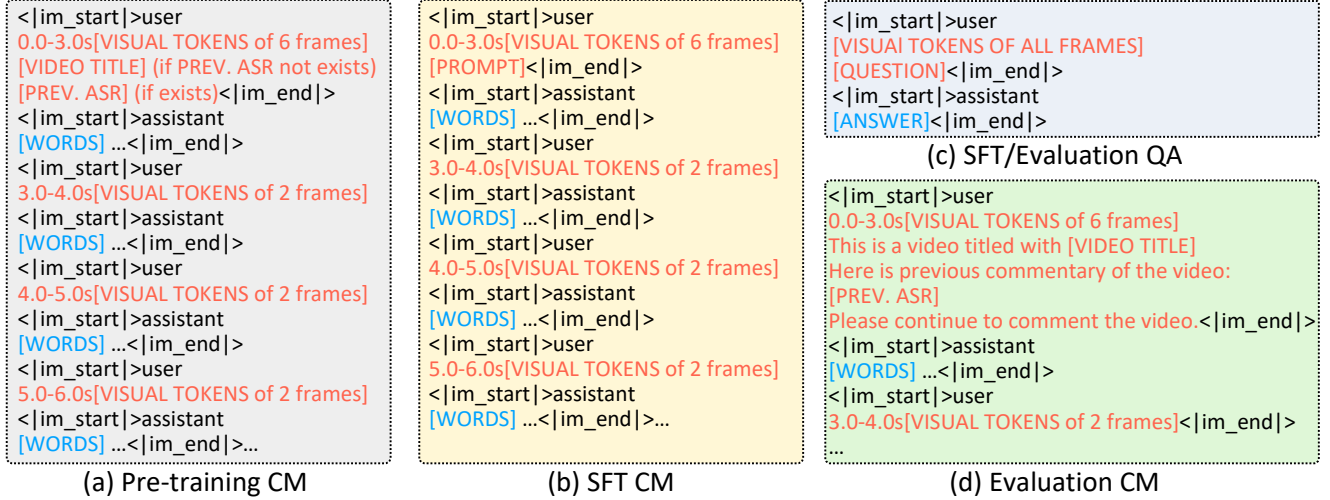


Figure 8. The prompts used during the pre-training instruction-tuning (aka. SFT) stages. CM represents commentary, QA denotes question-answering. For pre-training and instruction-tuning, the previous ASR texts are concatenated to form the context for the live commentary task if they are available. Otherwise, the context is formed by the video title. These contexts are masked during loss calculation. Note that QA data is incorporated exclusively during the instruction-tuning stage. As for inference, we remove the groundtruth in the prompts, *i.e.*, the words followed by a frame or the answer to a multiple-choice question.

Model	Latency	Input	Inf. Type
LLaVA-Video-72B [106]	20.51s	Clip	Captioning
LLaVA-Video-7B [106]	5.62s	Clip	Captioning
LiveCC-7B-Instruct	0.17s	Frame	Streaming

Table 5. The response latency comparison between LLaVA-Video-7/72B and our LiveCC-7B. Inf. is short for Inference.

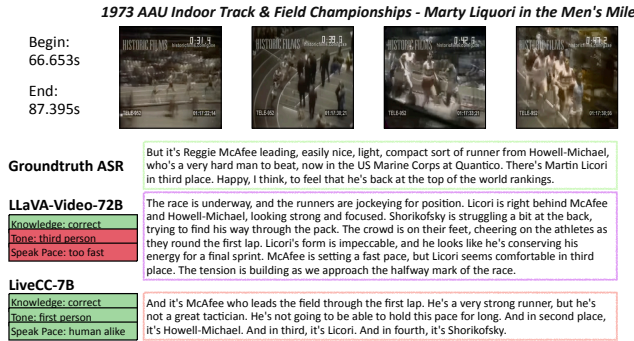


Figure 9. The comparison between the commentary generated by LLaVA-Video-72B and our LiveCC-7B-Instruct.

of the tested model and baseline text swapped.

9. Additional Experiments

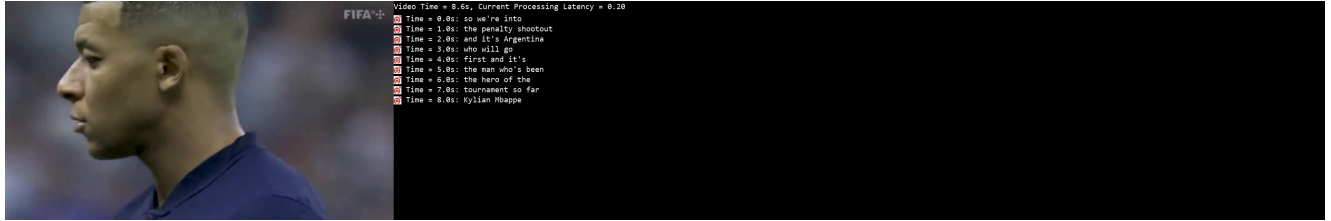
9.1. Response Latency

To highlight the efficiency of our streaming model, we present the response latency of LLaVA-Video-7B/72B

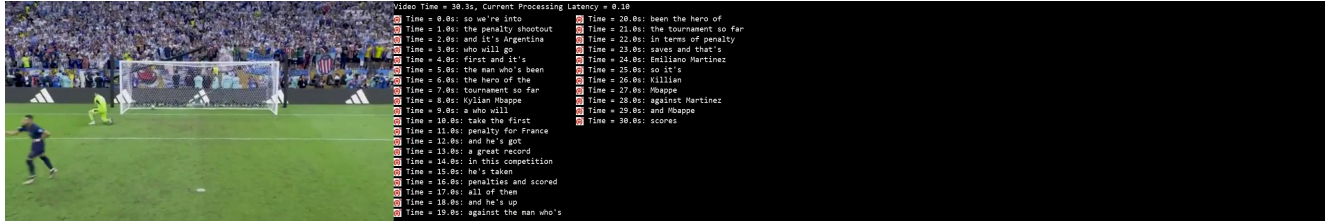
alongside our model in Table 5. Response latency is defined as the time a user waits to see the model’s output, a critical factor affecting user experience. Since the LLaVA-Video series are trained in a captioning style, requiring a full clip as input rather than a single frame, their response latency is significantly higher than that of our model. Notably, LiveCC not only achieves lower latency but also delivers high-quality commentary (see Table 4).

9.2. Commentary Quality

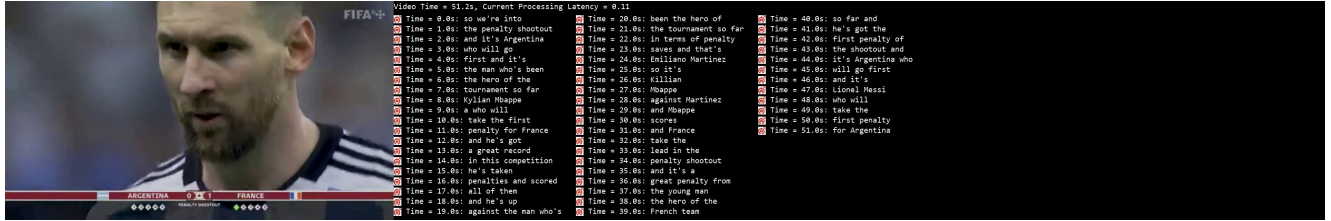
We analyzed the quality of the generated content, as shown in Figure 9. Benefiting from training on millions of ASR-transcribed videos, our model produces commentary that is more aligned with human preferences in terms of tone and speaking pace, while maintaining accurate event understanding. In contrast, the LLaVA-Video-72B, although capable of correctly describing the event, falls short in emulating human-like commentary.



(a) Video Time: 8.6s



(b) Video Time: 30.3s

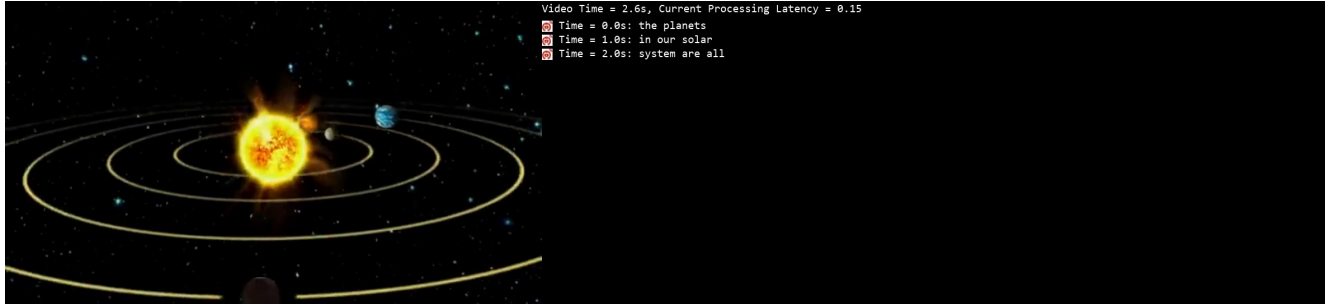


(c) Video Time: 51.2s

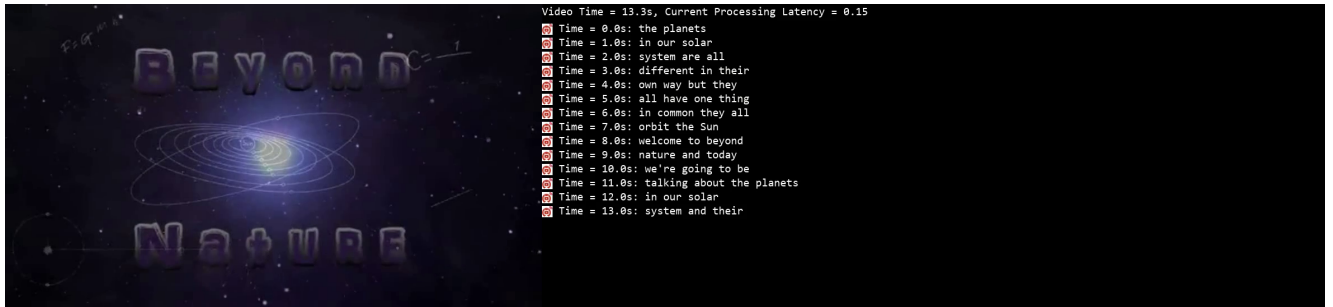


(d) Video Time: 77.2s

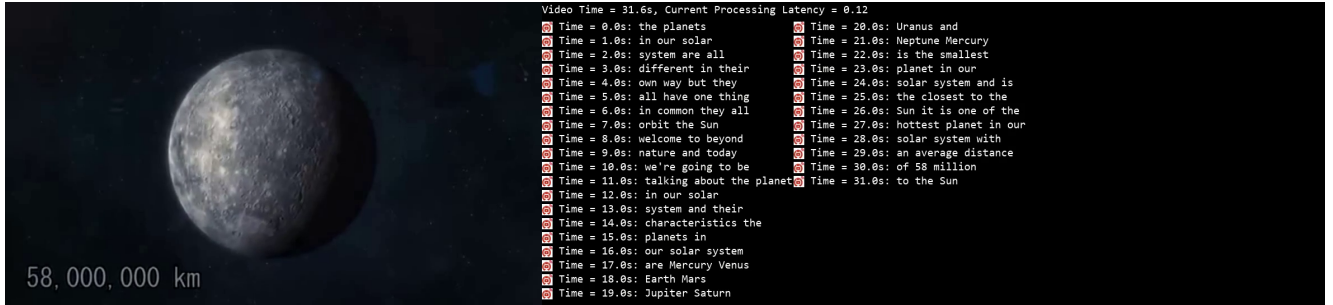
Figure 10. Real-time video commentary demo on unseen YouTube video (MCWJNOFJ0SM). The original YouTube title is “Argentina v France: Full Penalty Shoot-out — 2022 #FIFAWorldCup Final”. We only give a part of YouTube title “Full Penalty Shoot-out — 2022 #FIFAWorldCup Final” as prompt to avoid information leakage.



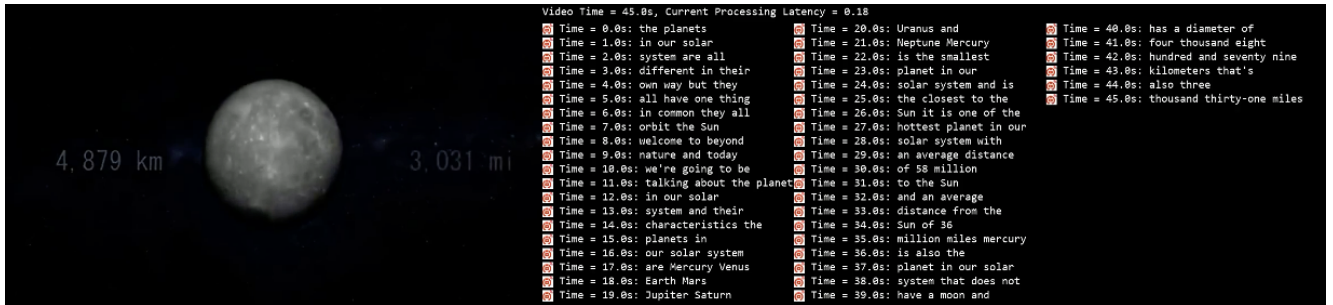
(a) Video Time: 2.6s



(b) Video Time: 13.3s

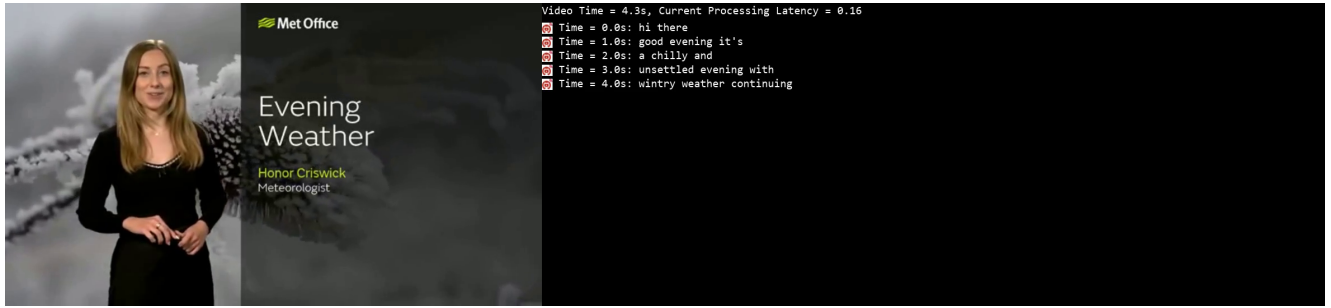


(c) Video Time: 31.6s



(d) Video Time: 45.0s

Figure 11. Real-time video commentary demo on unseen YouTube video (1cZTcfdZ3Ow). We give the YouTube title “The Planets In Our Solar System” as prompt.



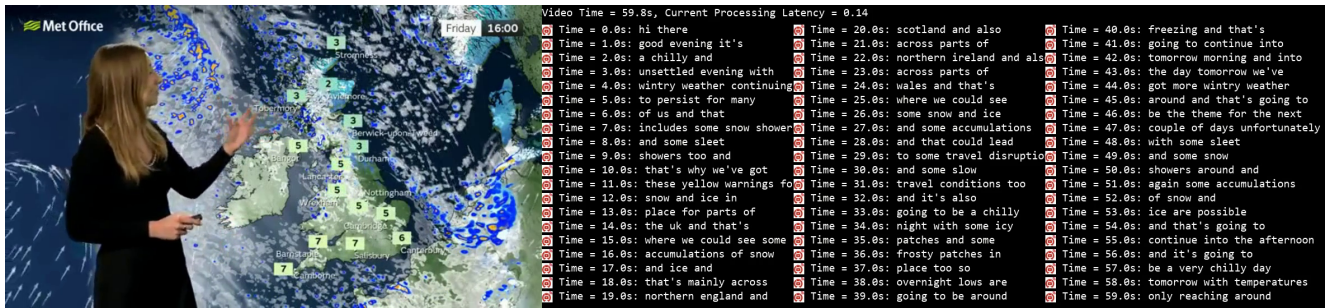
(a) Video Time: 4.3s



(b) Video Time: 24.8s

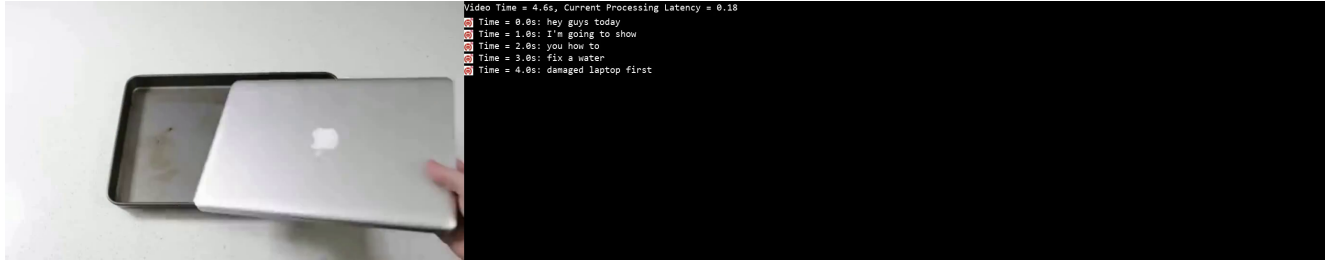


(c) Video Time: 43.8s

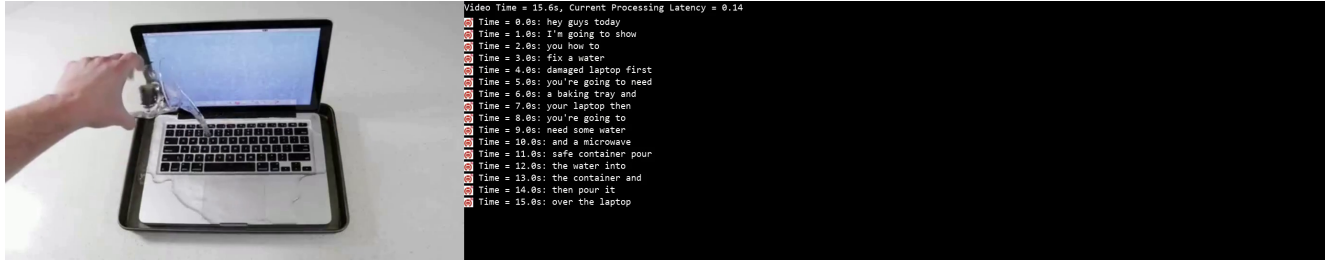


(d) Video Time: 59.8s

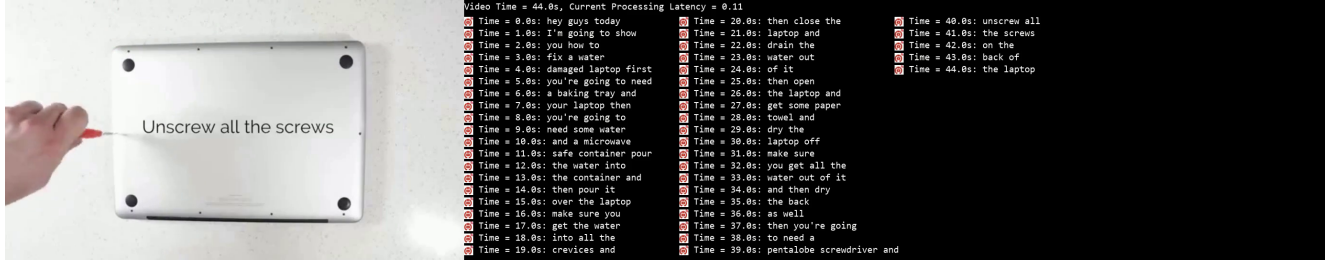
Figure 12. Real-time video commentary demo on unseen YouTube video (8XajZdrCDsk). The original YouTube title is “21/11/24 - Wintry weather perservering - Evening Weather Forecast UK – Met Office Weather”. We only give “21/11/24 - Wintry weather perservering” as prompt to avoid information leakage.



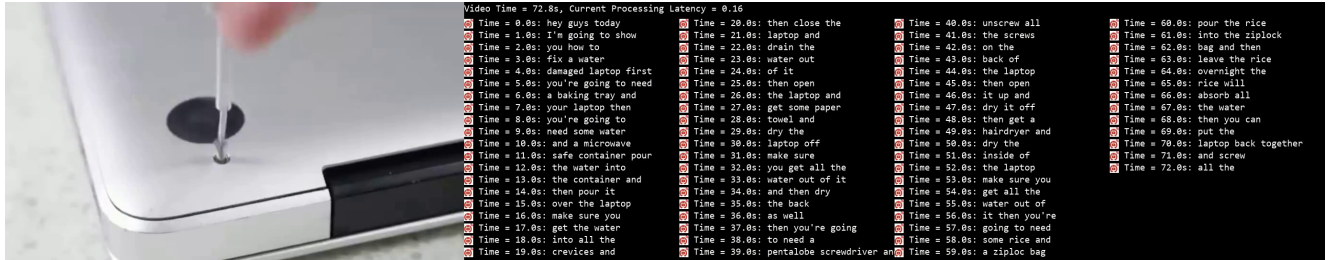
(a) Video Time: 4.6s



(b) Video Time: 15.6s



(c) Video Time: 44.0s



(d) Video Time: 72.8s

Figure 13. Real-time video commentary demo on unseen YouTube video (115amzVdV44). We give the YouTube title “How To Fix a Water Damaged Laptop” as the prompt.