LongVPO: From Anchored Cues to Self-Reasoning for Long-Form Video Preference Optimization

Zhenpeng Huang[1] **Jiaqi Li**[2] **Zihan Jia**[1] **Xinhao Li**[1,3] **Desen Meng**[1]
Lingxue Song[2] **Xi Chen**[2] **Liang Li**[2] **Limin Wang**[1,3,†]
[1]State Key Laboratory for Novel Software Technology, Nanjing University
[2]China Mobile Research Institute
[3]OpenGVLab, Shanghai AI Laboratory
https://github.com/MCG-NJU/LongVPO

Abstract

We present LongVPO, a novel two-stage Direct Preference Optimization framework that enables short-context vision-language models to robustly understand ultra-long videos without any long-video annotations. In Stage 1, we synthesize preference triples by anchoring questions to individual short clips, interleaving them with distractors, and applying visual-similarity and question-specificity filtering to mitigate positional bias and ensure unambiguous supervision. We also approximate the reference model's scoring over long contexts by evaluating only the anchor clip, reducing computational overhead. In Stage 2, we employ a recursive captioning pipeline on long videos to generate scene-level metadata, then use a large language model to craft multi-segment reasoning queries and dispreferred responses, aligning the model's preferences through multi-segment reasoning tasks. With only 16K synthetic examples and no costly human labels, LongVPO outperforms the state-of-the-art open-source models on multiple long-video benchmarks, while maintaining strong short-video performance (e.g., on MVBench), offering a scalable paradigm for efficient long-form video understanding.

1 Introduction

Recent vision-language models (VLMs)[31, 7, 57, 6, 39, 24] have demonstrated impressive capabilities in both image and video understanding. However, their performance often degrades when applied to tasks that require long-context visual reasoning, such as analyzing videos that span over an hour [45, 13, 59]. This presents a significant challenge in scaling VLMs for long-form video understanding.

While recent progress in long-video VLMs [41, 38, 6] has been encouraging, most approaches rely heavily on costly, high-quality annotations for long videos, limiting their scalability in practical applications. In contrast, existing short-context VLMs—despite being trained only on limited-frame inputs—have shown surprisingly competitive results on long-video benchmarks, largely thanks to their strong foundational vision-language alignment. This observation suggests a promising direction: short-context VLMs may possess untapped potential for long-video modeling if properly extended. This raises a natural question: *How far can we push short-context VLMs into the long-video regime—without the burden of expensive re-training or labels?*

To explore this question, we start with a strong short-context VLM [7, 19] that was not trained with long-range visual inputs and evaluate its performance on long-video understanding tasks. We identify two key challenges that limit its effectiveness: **(1) Scarcity of Long-Form Video Annotations:**

† Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

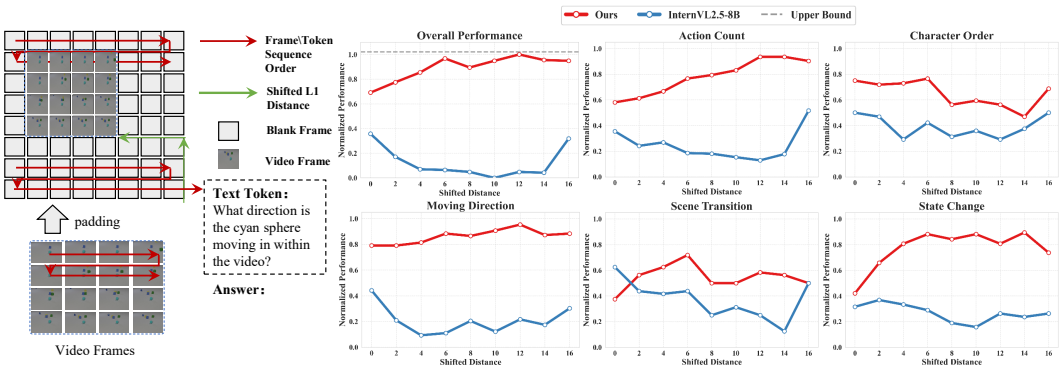


Figure 1: **Context Position Bias Probing. Left:** A short video segment (visualized as a 4×4 grid) is embedded within a much longer padded sequence and processed chronologically. **Right:** Performance is plotted against each frame's L1 distance from the question token. The middle-position drop indicates a strong positional bias ("lost-in-the-middle"). The Upper Bound shows performance without padding, revealing degradation under long-context settings.

High-quality video-text annotations, such as detailed captions or question-answer pairs, are typically available only for short clips, where annotators can reasonably cover the content. For long videos spanning tens of thousands of frames, such annotation becomes prohibitively expensive [51] and often suffers from incomplete coverage and poor temporal alignment [44]. **(2) Context-Length Bias in Short-Context VLMs:** Common practices such as YARN [34], NTK can extend positional encodings for longer sequences, but the resulting models still suffer from position-related biases and limited performance gains. To simulate long-context scenarios, we embed the original short video's 1D frame sequence within a much longer sequence, padding the surrounding positions with blank frames (visualized as a grid in Fig. 1). This setup allows us to investigate the model's sensitivity to spatial positions across extended contexts. Specifically, by computing the L1 distance between each frame's position and a fixed query point, we uncover a "lost-in-the-middle" phenomenon—analogous to what has been observed in long-sequence language models [3]—where the model's performance dips for inputs located near the center of the grid. This highlights a positional bias that disfavors centrally located content (see Fig. 1).

Recent efforts [22] attempt to address this by leveraging Direct Preference Optimization (DPO) to enhance grounding capabilities. However, as depicted in Fig. 2, this method assumes access to a reference model that already supports long-context reasoning. This assumption does not hold for short-context VLMs. Moreover, this approach requires proprietary models to generate and filter preference data, which introduces external language model biases without fully viewing the video. As a result, the method fails to fundamentally resolve the problem and delivers suboptimal performance.

To address these challenges, we propose a two-stage training framework that extends short-context VLMs to ultra-long video contexts, as shown in Fig. 3: **Stage 1: Efficient Short-to-Long Learning from Anchored Cues.** We form mixed, interleaved sequences of short clips from the SFT dataset. For each clip, we generate an anchor question and use the short-context VLM's answer as the Preferred Response, ensuring via scalable auto-filtering that each question refers to exactly one clip. The model learns to maximize the likelihood of the Preferred Response given the anchor question and its corresponding clip. To simulate distracting contexts, we introduce Dis-Preferred Responses by prompting temporally misaligned clips, forcing the model to retrieve the correct answer from many candidates. We also randomize the target clip's position within the sequence to mitigate positional biases during training. **Stage 2: Self-Training for Long Video Preference Alignment.** Building on Stage 1's memory and retrieval single-segment skills, we train the model to handle longer, more complex videos, without requiring ground-truth annotations. First, we employ a recursive captioning pipeline to generate structured, scene-level metadata to leverage the model's short-context capabilities. We then transfer insights from open-source LLMs' long-text understanding: given a sequence of scene-organized captions, we generate questions and identify the minimal set of scenes needed to answer them. We then craft Dis-Preferred Responses by prompting the model with partial or misleading context (e.g., omitting critical scenes), encouraging it to assemble the complete information chain required for accurate answers in real-world, long-video scenarios.

In summary, our contributions are threefold:

- We introduce LongVPO, a two-stage framework that extends short-context VLMs to long video contexts without relying on any long-video annotations.
- We construct a synthetic DPO training set from short visual context transfer to long text context for long videos, using only ~16k instances—significantly fewer than existing instruction-tuning datasets —eliminating the need for long-video labels.
- Our approach outperforms existing long-video models trained on large-scale supervised and preference-optimized data across challenging long-video understanding benchmarks, while maintaining competitive performance on short-video tasks, offering a new superior paradigm for efficient multimodal long video understanding.

2 Related work

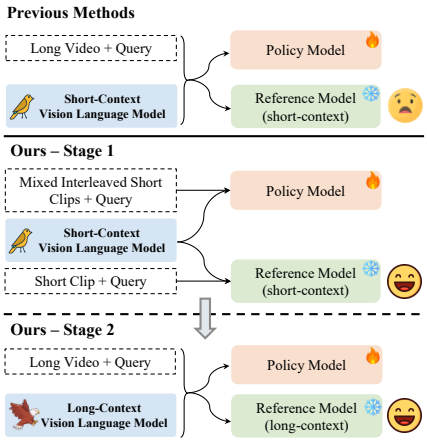


Figure 2: Comparison of prior methods with our proposed two-stage method.

VLMs for Long Video Understanding. Advancements in vision-large models (VLMs) [7, 10, 18, 20, 25, 30, 57] have shown impressive capabilities in video understanding, and many models demonstrated excellent performance in short video analysis. Some recent works make further efforts towards long video understanding by designing certain strategies to compress/select visual context [9, 12, 16, 21, 38, 40] or extend the temporal context window [6, 39, 54]. In addition to the innovation of model architecture, it is also important to construct long-form video instruction datasets to guide VLM to extract detailed visual cues and model cross-temporal relationships. Representative methods such as VideoChat-Flash [24], Kangaroo [26], and Video-XL [40] build data production pipelines and curate long video data to enhance the ability of video VLM. However, high-quality long video annotations could be expensive and time-consuming, making it difficult to obtain and scale up [5] compared to short videos. Therefore, how to develop an efficient strategy to employ short video-text data to facilitate VLMs in long video understanding remains a challenge. In this work, we propose a novel two-stage framework for VLMs to progressively learn the ability to analyze longer videos using only short video annotations.

DPO for Video-VLMs. As a post-training strategy, DPO has been frequently adopted in the development of VLMs [14, 27, 46, 58, 60]. Unlike the next-token prediction used in the SFT step, DPO refines the VLM using triplets of queries, preferred and rejected responses, reducing model hallucination and better aligning with human reasoning [15]. The simplicity and strong performance of DPO training further encourage researchers to apply it to video-based VLMs, where devising effective spatial and temporal perturbation tasks is a crucial part [2, 15, 17, 22, 23, 52, 55]. Recent works propose methods such as frame cutout, spatial misalignment, clip dropping, clip rearrangement and frame disconnection to generate query and corresponding preference responses, with the help of proprietary or open-sourced models [15, 22, 23, 52]. The curated data are employed in the VLM training to enhance the spatial-temporal perception and dynamic modeling capabilities for videos. While these efforts have shown promise, many existing works focus on minute-level or short-form videos. For long visual context, directly generating preference data remains challenging due to task design complexity and high computation costs, a limitation even in recent explorations of DPO for long videos [22, 23]. Therefore, our progressive DPO training that incrementally extends the model's capacity to capture long temporal dependencies may offer a more practical and scalable path towards long video understanding.

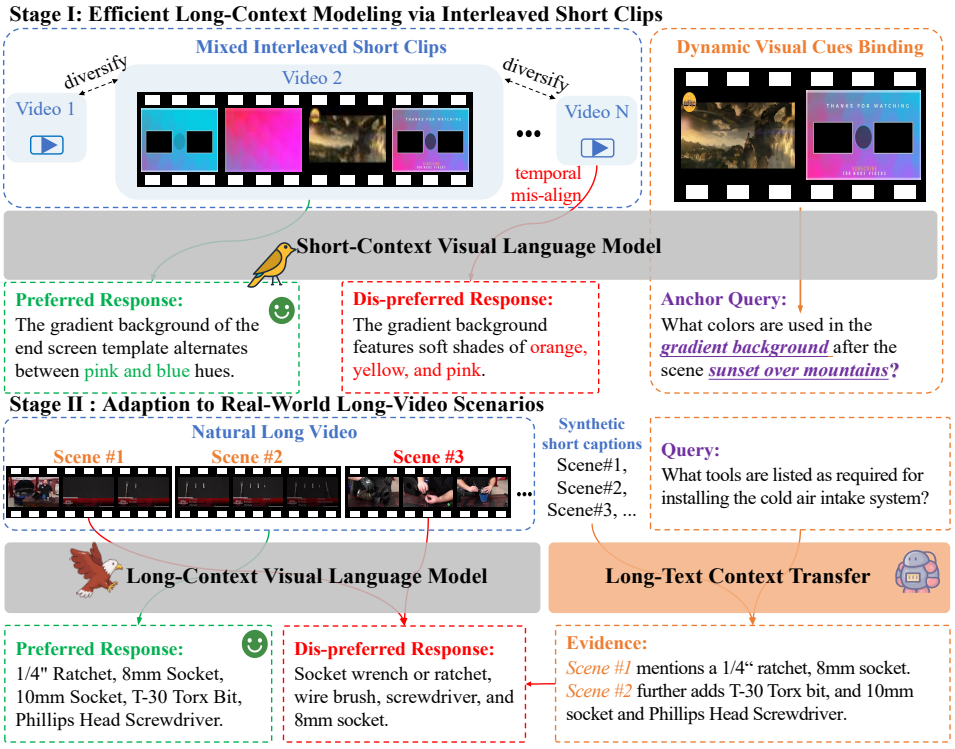


Figure 3: Overview of our two-stage training framework. **Stage 1:** Short clips from an SFT dataset are interleaved into a long composite video. The ground-truth annotation of an *anchor* clip provides the query (q_i) and preferred response (y_i^+), while dispreferred responses (y_i^-) are generated using content from distractor clips. Samples are filtered via scene-similarity (e.g., DINOv2) to ensure unambiguous supervision. **Stage 2:** A recursive captioning pipeline produces scene-level metadata for unlabeled long videos. A long-context LLM then generates the query, reasoning trace, and preferred response, while dispreferred responses are created by prompting with partial context.

3 Method

3.1 Background

Direct Preference Optimization (DPO) [35] aligns a policy model π_θ with human preferences by directly optimizing a policy that best satisfies the preferences, using a simple classification loss. The objective function is formulated as maximizing the log-sigmoid of the log-likelihood ratio between preferred (y_i^+) and dispreferred (y_i^-) responses, relative to a frozen reference model π_{ref}:

$$\mathcal{L}_{\text{DPO}}(\theta) = -\sum_i \log \sigma \left[\beta \left(\log \frac{\pi_\theta(y_i^+ \mid x_i)}{\pi_{\text{ref}}(y_i^+ \mid x_i)} - \log \frac{\pi_\theta(y_i^- \mid x_i)}{\pi_{\text{ref}}(y_i^- \mid x_i)} \right) \right], \tag{1}$$

where σ denotes the sigmoid function, and $\beta > 0$ is a hyperparameter that controls the strength of the preference margin, effectively determining how sharply the policy should prefer y_i^+ over y_i^-.

3.2 Distribution Shift from Short to Long Video Contexts

Consequently, directly applying a model fine-tuned on short-video data to preference optimization tasks in long videos reveals two critical challenges:

- **Reference-model degradation in extended contexts:** As depicted in Fig. 1, the short-context visual model not only exhibits position bias but also shows over-specialization to short temporal spans, limiting its generalization to extended video contexts. Therefore, in the DPO framework, the reference model π_{ref} is typically kept frozen. As the policy model π_θ is trained to understand and leverage long-range dependencies inherent in extended

video sequences, its output characteristics for queries over these long videos may diverge substantially from the short-video patterns that π_{ref} was trained to evaluate. This divergence can impair the ability of π_{ref} to serve as a consistent and meaningful baseline in the DPO objective, potentially leading to unstable or suboptimal policy updates when adapting to long-video tasks.

- **Annotation scarcity and distribution gap:** The lack of dense, high-quality annotations for long video sequences creates a significant distributional disparity between the data available for initial supervised fine-tuning (SFT) or reference model training (typically short clips) and the target domain of long videos. This data gap can severely degrade model performance when generalizing to longer contexts.

3.3 Stage 1: Efficient Short-to-Long Learning from Anchored Cues

To mitigate the distribution shift from short to long video contexts and effectively leverage abundant short-video SFT data without compromising performance on short clips, we design an anchor-based approach that preserves short-context fidelity while exposing the model to long-range contextual variation. This strategy encompasses three core components: the synthesis of anchor-centric preference triples, their subsequent refinement through filtering, and a specific adaptation of the DPO objective to robustly incorporate this data.

The synthesis of preference data forms the initial component. At its core is the generation of preference triples (q_i, y_i^+, y_i^-) where the query q_i is answerable only from a designated "anchor" short clip within a longer composite video x_i. This *Dynamic Visual Cues Binding* process involves three key steps:

- **Anchor-centric QA and Preferred Response Generation:** A short video clip, potentially with supplementary annotations (e.g., captions), is randomly selected from SFT data to serve as the anchor $x_{i,\text{anchor}}$. A question-answer (QA) pair (q_i, y_i^+) is then generated by the target short VLM such that q_i can only be answered comprehensively using information present exclusively within $x_{i,\text{anchor}}$. The answer y_i^+ constitutes the preferred response.
- **Composite Sequence Assembly:** Multiple distinct short clips, including the designated anchor $x_{i,\text{anchor}}$, are concatenated to form the longer composite video sequence $x_i = [x_{i,1}, ..., x_{i,\text{anchor}}, \ldots, x_{i,k}]$.
- **Plausible Dispreferred Response Generation:** For the same question q_i, a dispreferred response y_i^- is generated. This response is designed to be plausible yet incorrect, typically by drawing information from non-anchor clips within x_i to simulate anchor positioning errors.

Following data synthesis, a critical step is to ensure the quality and unambiguity of the preference signals. To guarantee that the anchor clip $x_{i,\text{anchor}}$ is genuinely unique in containing the necessary information to answer q_i, we introduce two complementary post-filtering mechanisms:

- **Scene-Similarity Filtering:** We extract per-clip visual embeddings using a robust vision encoder (e.g., DINOv2 [32]). Any non-anchor clips within x_i exhibiting an embedding similarity to $x_{i,\text{anchor}}$ above a predefined threshold are replaced, or the entire sample is discarded. This enforces greater visual dissimilarity between the anchor and distractor segments.
- **Question Specificity Filtering:** A capable large language model (e.g., Qwen-2.5 32B [47]) is prompted to verify that q_i necessitates reference to at least two to three distinct visual elements (e.g., specific objects, attributes, or events) present or occurring in $x_{i,\text{anchor}}$. Questions failing this specificity test, indicating they could potentially be answered by other clips, are discarded.

As discussed previously, applying the short-clip reference model π_{ref} to the full input x_i leads to performance degradation due to context-length mismatch. To address this, we introduce an **anchor-only approximation**, which leverages the design hypothesis that only the anchor clip $x_{i,\text{anchor}}$ contains information necessary to answer q_i, while non-anchor segments provide no relevant signal. This hypothesis is supported by our filtering process, which reduces semantic similarity between anchor and non-anchor clips, reinforcing the anchor's informational sufficiency.

Under this approximation, the reference model's likelihood is evaluated solely on the anchor clip:

$$\pi_{\text{ref}}(y \mid x_i) \approx \pi_{\text{ref}}(y \mid x_{i,\text{anchor}}). \tag{2}$$

This avoids context-length mismatch, reduces computational and memory costs, and ensures likelihoods reflect only anchor-related content. The modified DPO objective thus becomes:

$$\mathcal{L}_{\text{stage1}}(\theta) = -\sum_i \log \sigma \left[\beta \left(\log \frac{\pi_\theta(y_i^+ \mid x_i)}{\pi_{\text{ref}}(y_i^+ \mid x_{i,\text{anchor}})} - \log \frac{\pi_\theta(y_i^- \mid x_i)}{\pi_{\text{ref}}(y_i^- \mid x_{i,\text{anchor}})} \right) \right]. \tag{3}$$

3.4 Stage 2: Self-Training for Long Video Preference Alignment

While the Efficient Short-to-Long Video Alignment method scales input length via synthetic clip compositions, such sequences often lack the natural coherence and narrative structure of genuine long videos, which is crucial for preference learning that depends on temporally grounded reasoning and causal event understanding. This becomes problematic for queries requiring temporal reasoning (e.g., action chains or evolving events). To bridge this gap, we propose a self-training framework for aligning long-video preferences.

Data Preparation. In this stage, we first employ a recursive captioning strategy to generate dense textual descriptions for long videos. For each temporally segmented scene within a long video, the target model is conditioned on both the current video segment and the captions generated for preceding scenes. This iterative process constructs a coherent, context-aware caption sequence for the entire video, capturing local semantics and their broader contextual dependencies.

Construction of Preference Data (q_i, y_i^+, y_i^-) for Self-Training. The generation of preference triples for the self-training stage involves a multi-step process, leveraging both a Large Language Model (LLM) for query and reasoning articulation, and the target Multimodal Large Model (MLLM) itself for generating preferred responses.

1. **Long Text Context Knowledge Transfer.** Query and Reasoning Generation by LLM: Given the long video content (represented by its scenes and recursive captions), the LLM is prompted to produce a pair (q_i, r_i), where q_i is a user-style query about the video content, and r_i is a detailed, multi-step reasoning trace that explicitly references unique scene identifiers (e.g., "Scene #N") as **binary scene-question relevance labels**.
2. **Preferred Response (y_i^+) Generation by the Target MLLM**: Specifically, for each query q_i (obtained from the LLM in the previous step) and the corresponding full long video x_i, the MLLM π_θ is prompted to generate a response. This directly generated output from $\pi_\theta(y \mid q_i, x_i)$ is designated as the preferred response y_i^+ for the DPO objective. This approach uses the MLLM's current capabilities to articulate what it deems a good response to the query based on the video.
3. **Dispreferred Response (y_i^-) Generation**: Dispreferred responses y_i^- are generated by introducing specific reasoning errors into the original trace r_i. We employ two error patterns to construct flawed but plausible responses: *1) Reasoning from Partial Evidence:* The target MLLM is prompted to generate a dis-preferred response based on only a subset of the scenes detailed in r_i as essential for a comprehensive answer. *2) Ignoring Critical Evidence:* The target MLLM generates a dis-preferred response that omits references to one or more critical scenes from r_i or improperly focuses on irrelevant scenes. The resulting flawed reasoning traces serve as dispreferred responses y_i^-.

For Stage 2, we employ the standard DPO objective $\mathcal{L}_{\text{stage}_2}(\theta) = \mathcal{L}_{\text{DPO}}(\theta)$. While the self-generated y_i^+ may not be perfect, the relative preference delta (y_i^+, y_i^-) provides a valid training signal. The policy model π_θ is initialized from the Stage 1 checkpoint, and the reference model π_{ref} is frozen as the Stage 1 checkpoint, which Stage 1 equipped with the basic capability to retrieve query-relevant clips from the full long-video input.

3.5 Total Objective

In both stages $i = 1, 2$, we incorporate the SFT loss into the DPO framework, weighted by α following [33]. The total objective is defined as:

$$\mathcal{L}(\theta) = \mathcal{L}_{\text{stage}_i}(\theta) + \alpha \cdot \frac{-\log \pi_\theta(y^+ \mid x)}{|y^+|}, \tag{4}$$

Models	Size	Frames	LVBench	LongVideoBench	MLVU	Video-MME (wo / w sub)		MVBench
			Overall	Validation	M-Avg	Overall	Long	Overall
Average Duration			4101s	473s	651s	1,010s	2,386s	16s
Proprietary Vision-Language Models								
GPT4-V [1]	*Undisclosed*		-	*59.1*	*49.2*	*59.9 / 63.3*	*53.5 / 56.9*	43.7
GPT4-o [31]	*Undisclosed*		*30.8*	*66.7*	*64.6*	*71.9 / 77.2*	*65.3 / 72.1*	64.6
Gemini-1.5-Pro [37]	*Undisclosed*		*33.1*	*64.0*	-	*75.0 / 81.3*	*67.4 / 77.4*	60.5
Open-Source Multi-Image Vision-Language Models								
LLaVA-OneVision [19]	72B	32	-	61.3	66.4	66.3 / 69.6	60.0 / 62.4	59.4
InternVL2 [8]	76B	16	-	61.0	69.9	61.2 / 67.8	-	69.6
LLaVA-OneVision [19]	7B	32	-	56.5	64.7	58.2 / -	-	56.7
Oryx-1.5 [28]	7B	128	-	56.3	67.5	58.8 / 64.2	-	-
MiniCPM-v2.6 [49]	8B	64	-	54.9	37.3	60.9 / 63.7	51.8 / 56.3	-
mPLUG-Owl3 [50]	7B	16	-	52.1	63.7	59.3 / -	50.1 / -	-
Qwen2-VL [47]	7B	2FPS	-	55.6	-	63.3 / 69.0	-	67.0
NVILA [29]	7B	256	-	-	70.1	64.2 / 70.0	54.8 / 63.3	-
Open-Source Video-Language Models								
VideoLLaMA2 [10]	72B	16	-	-	61.2	62.4 / 64.7	57.6 / 59.0	-
LLaVA-Video [57]	7B	1FPS	-	58.2	70.8	63.3 / 69.7	-	58.6
Video-XL [40]	7B	2,048	-	49.5	64.9	55.5 / 61.0	49.2 / -	-
VideoLLaMA2 [53]	7B	16	-	-	48.5	47.9 / 50.3	-	-
Video-CCAM [12]	9B	96	-	-	58.5	53.2 / 57.4	46.7 / 49.9	-
Kangaroo [26]	8B	64	-	54.8	61.0	56.0 / 57.6	46.7 / 59.3	61.0
LongVU [38]	7B	1FPS	-	-	65.4	60.6 / 59.5	-	66.9
LongVA [54]	7B	128	-	-	56.3	52.6 / 54.3	46.2 / 47.6	-
LongVILA [6]	7B	256	-	57.1	-	60.1 / 65.1	-	67.1
VideoChat-Flash [24]	7B	512	48.2	64.7	74.7	65.3 / 69.7	55.4 / 63.3	74.0
InternVL2.5 [7]	8B	64	43.2	60.0	68.9	64.2 / 66.9	-	72.0
InternVL2.5 [7]	8B	512	45.2	62.7	67.6	61.1 / 65.3	51.1 / 57.2	72.0
+LongVPO (128f)								
Stage1	8B	512	**49.4** (+4.2)	**65.4** (+2.7)	**73.5** (+5.9)	**64.2** (+3.1) / **70.1** (+4.8)	**53.8** (+2.7) / **62.8** (+5.6)	**72.9** (+0.9)
Stage2	8B	512	**50.1** (+4.9)	**66.6** (+3.9)	**74.1** (+6.5)	**64.6** (+3.5) / **70.3** (+5.0)	**55.3** (+4.2) / **64.2** (+7.0)	**73.1** (+1.1)
+LongVPO (256f)								
Stage1	8B	512	**49.6**(+4.4)	**66.0**(+3.3)	**74.8**(+7.2)	**65.0**(+3.9) / **71.2**(+5.9)	**55.8**(+4.7) / **65.1**(+7.9)	**72.9**(+0.9)
InternVideo2.5 [43]	8B	512	47.4*	63.2*	72.8	63.3* / 71.1*	52.6* / 65.1*	75.7
+LongVPO (256f)								
Stage1	8B	512	**50.9** (+3.5)	**67.0** (+3.8)	**74.0** (+1.2)	**65.4** (+2.1) / **72.6** (+1.5)	**54.3** (+1.7) / **67.0** (+1.9)	**74.7** (-1.0)
Stage2-iter1	8B	512	**51.0** (+3.6)	**67.2** (+4.0)	**74.4** (+1.6)	**65.6** (+2.3) / **72.5** (+1.4)	**54.9** (+2.3) / **67.1** (+2.0)	**75.1** (-0.6)
Stage2-iter2	8B	512	**51.0** (+3.6)	**67.2** (+4.0)	**74.7** (+1.9)	**66.1** (+2.8) / **73.1** (+2.0)	**56.1** (+3.5) / **67.4** (+2.3)	**75.1** (-0.6)
+LongVPO (512f)								
Stage1	8B	512	**50.4** (+3.0)	**67.8** (+4.6)	**75.0** (+2.2)	**65.6** (+2.3) / **73.0** (+1.9)	**54.9** (+2.3) / **67.1** (+2.0)	**75.1** (-0.6)

Table 1: Accuracy (%) on the short and long video understanding benchmarks. **Size** indicates the number of parameters. **Frames** denotes the maximum number of frames sampled from each video or the frame sampling rate (FPS). The best and second-best results among open-source models of similar size (7∼9B) are in **bold** and underlined, respectively. "256f"/"512f" refer to the maximum number of training frames. * denotes reproduced results.

where $\mathcal{L}_{\text{stage}_i}(\theta)$ denotes the DPO loss at stage i, $\pi_\theta(y^+ \mid x)$ represents the model likelihood of the preferred response y^+.

4 Experiment

4.1 Implementation Details

Baseline. We adopt InternVL-2.5-8B [7] as the base model of our framework. It comprises InternViT-300M as the vision encoder and InternLM-2.5-7B-32K [4] as the language backbone. According to the official report, the model was trained on a maximum of approximately 32 video frames, corresponding to a visual context length of around 8192 tokens. We implement DeepSpeed Ulysses sequence parallelism to enable efficient training with 32K extended video context length.

Data Preparation. To ensure a fair and leakage-free evaluation, we rely solely on publicly available datasets. Specifically, Stage 1 training utilizes caption annotations from LLaVA-Video-178K [56]. For stage 2, we incorporate scene-segmented but unlabeled long videos from Vript [48], both of which are included in the InternVL-2.5 SFT dataset.

For Stage 1 data, we preprocess each source clip from LLaVA-Video-178K by uniformly sampling up to 64 frames at 1 fps. To construct each composite training instance x_i, we designate one clip as the *anchor video* ($x_{i,\text{anchor}}$) and randomly select several additional clips as *non-anchor video*. Any non-anchor video whose DINO embedding cosine similarity with the anchor exceeds 0.6 is discarded.

Video Understanding Benchmarks. To comprehensively evaluate our model's long-context understanding capabilities, we adopt three existing long-video benchmarks [42, 45, 59] along with the

comprehensive benchmark VideoMME [13]. Additionally, we verify performance on MVBench [20]. The evaluated video durations from all these benchmarks span from a few seconds to 2 hours.

4.2 Main Results

Our main results are shown in Tab. 1. **LongVPO exhibits strong competitiveness among models of comparable scale on long-context video understanding benchmarks.** Notably, most competing long-video models are trained on datasets with manual curation [11, 36] or rely on proprietary MLLMs [31, 37] to annotate. In contrast, our approach leverages only around 16K synthetic samples, without any reliance on expensive human annotations or closed-source tools, underscoring the effectiveness of our training strategy and data construction methodology.

LongVPO maintains strong performance in short-video analysis. Although our primary goal is to improve long-video understanding, our model also achieves competitive results on the general-purpose short-video benchmark MVBench, even surpassing prior results with a +1.1 improvement. This further illustrates the effectiveness of LongVPO in accommodating videos of varying durations in real-world scenarios.

Effectiveness of Stage 1. In Stage 1, the model is trained on synthetic long-video data, resembling a form of structureless memory training. The primary goal is to mitigate context bias and activate the model's localization ability. As shown in Tab. 1, the model generalizes well to real-world scenarios and the objectives are largely fulfilled.

Effectiveness of Stage 2. Instead of Stage 1 focusing on localizing a single segment to answer questions, Stage 2 focuses on aggregating information across multiple segments from real long videos, thereby enhancing the model's capacity to extract complex, question-relevant content. We observe consistent improvements in most settings, indicating that training on real video domains results in better alignment with realistic scenarios.

Long Video Context Evaluation. Following the NIAH setup from LongVA [54], we densely sample multiple frames from a long video and insert a selected image at different positions within the sampled frames. We evaluate InternVL2.5 on a maximum of approximately 3k frames. The results in Fig. 4 show that the Baseline Model begins to exhibit significant performance degradation at around 800 frames, with complete failure to follow instruction output formats when reaching approximately 1k frames, while LongVPO demonstrates superior long-context modeling capabilities.

4.3 Extending to Dedicated Long-Context Models

While the primary results are based on short-video models, we further validate the generalizability of our approach by applying it to InternVideo2.5 [43], a representative long-video model. InternVideo2.5 incorporates two key designs for long-context understanding: (1) pre-training on high-quality long-video datasets, supporting up to 256 frames during training; (2) a specialized redundancy compression mechanism in its vision-language connector for efficient long-context processing.

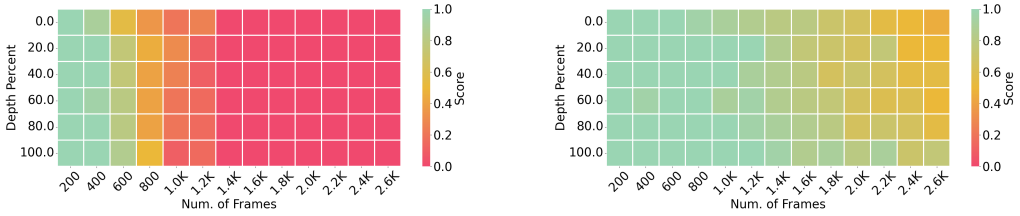
Generality beyond Short-Context Models Although initially designed under short-video assumptions, our method demonstrates strong generalization. When applied to InternVideo2.5 for continued training, LongVPO consistently surpasses its counterparts trained on InternVL2.5. This indicates that InternVideo2.5 has not yet saturated in long-context understanding, and our approach provides further enhancement, underscoring its adaptability even for models pre-trained on long videos.

Context Length Scalability We evaluate the scalability of our approach under varying context lengths. For Stage 1, we extend the length of the synthesized video by simply selecting more clips. Our approach consistently improves performance as the maximum number of input frames increases (e.g., from LongVPO-256f to LongVPO-512f), demonstrating strong scalability and an enhanced ability to exploit longer temporal contexts.

4.4 Ablation Study

4.4.1 Data Preparation.

Scene Filtering in Stage 1. As shown in Tab. 2, we conduct ablations on scene filtering and response selection. In Stage 1, removing the scene filter—particularly when relying on simple Top-K



(a) Baseline Model with 0% acc in around 1k Frames.

(b) LongVPO(Ours)

Figure 4: The V-NIAH results of our baseline InternVL2.5-8B and LongVPO. "Frame Depth" indicates the position where the needle image is located, ranging from 0% to 100% (from the beginning to the end of the video).

selection—results in performance degradation. This underscores the importance of semantic filtering for robust long video understanding.

Chosen Response in Stage 2. In Stage 2, we compare three approaches for generating the chosen response: the target VLM directly processing the original video to produce responses (self-generated), responses generated by Qwen2.5-32B as used in our long-text context transfer method (Qwen-32B selected), and the target VLM receiving a combination of video frames and synthetic captions as input (interleaved). Surprisingly, the latter two methods yield suboptimal performance, highlighting the efficiency of our training process with only scene-question relevance labels.

LLM Backbone in stage2. LongVPO maintains strong performance even when using the 7B parameter InternLM2.5 as its backbone instead of Qwen2.5-32B, with only a slight drop observed. This demonstrates that its effectiveness is not dependent on a larger LLM.

4.4.2 Baseline Comparison

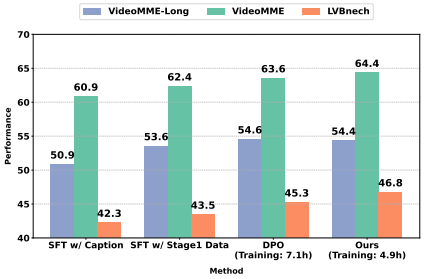


Figure 5: Comparison of Stage 1 training using SFT and DPO. Additional results are provided in the appendix.

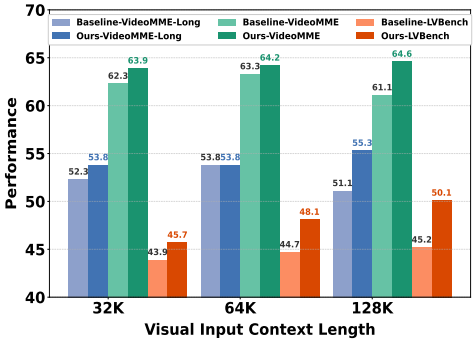


Figure 6: LongVPO consistently outperforms the baseline (InternVL2.5-8B), with performance gains increasing as more input frames are used, while the baseline plateaus.

Scaling with Input Frames. As the number of input frames increases, LongVPO exhibits progressively larger performance gains, whereas the baseline model (InternVL-2.5-8B) shows signs of saturation. As shown in Fig. 6, LongVPO achieves superior performance across benchmarks, in contrast to the baseline, which stagnates with longer contexts. These results confirm that LongVPO can more effectively extract and utilize temporal information from extended video sequences, highlighting its advantage in long-video understanding.

Stage 1: SFT vs. DPO. As illustrated in Fig. 5, fine-tuning with single-video caption data leads to a noticeable performance drop. We attribute this to overfitting, since these captions were already part of the SFT dataset. Applying DPO to the same captions yields significant improvements, suggesting that the DPO framework enables more data-efficient learning and mitigates overfitting.

Stage 1: DPO with Long Videos. Consistent with the observation in Fig. 1, using complete long videos as input to the reference model leads to suboptimal outcomes. Our ablation study confirms that

Table 2: Ablation study on scene filtering and response selection methods across all long-video benchmarks. MLVU, LongVideoBench, and LVBench use a 32K context window during inference.

Stage	Setting	MLVU	LongVideoBench	LVBench
Stage 1	w/ Scene Filter	**72.9**	**66.1**	**45.3**
	w/o Scene Filter (adding a similar one)	69.8	64.2	43.4
	w/o Scene Filter (TopK)	69.9	58.4	–
Stage 2	*Chosen response Choice*			
	Self-generated response	72.9	**66.1**	**45.3**
	LLM-generated (Qwen2.5-32B) response	**73.1**	65.6	44.4
	Self-generated w/ scene-interleaved caption	73.0	66.1	44.7
	Long Context Knowledge Transfer Backbone			
	InternLM2.5-7B instead of Qwen2.5-32B	72.5	65.8	44.9

this approach is less effective than our proposed method on both general and long-video benchmarks, while our method requires only approximately 70% of its training time.

4.4.3 Qualitative Comparison

Fig. 7 evaluates long video understanding through a pumpkin-carving action counting task. This challenging benchmark requires both action recognition and temporal instance tracking across extended durations. Current strong multimodal models (Qwen2.5-VL, Qwen2-VL, LLaVA-Video) failed to provide correct counts, while our LongVPO accurately identified all 5 instances. This demonstrates LongVPO's superior temporal understanding and counting capability in long videos, outperforming otherwise powerful baselines in complex comprehension tasks.

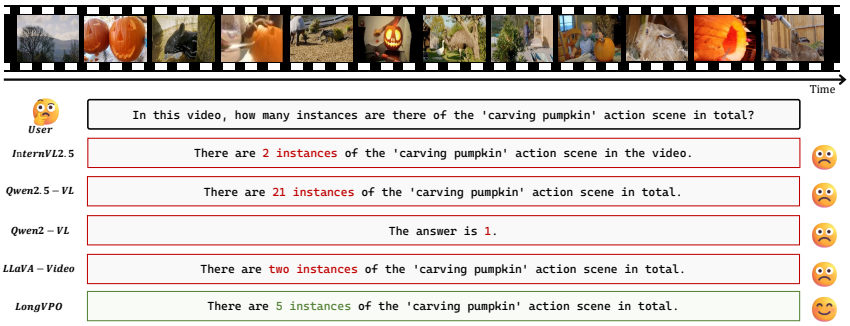


Figure 7: Qualitative comparison on long video understanding. More details are in the appendix.

5 Conclusion

We propose LongVPO, a novel two-stage DPO training framework tailored for long video understanding. By leveraging only 16k synthetic DPO instances constructed from short visual contexts, our method incrementally extends the capabilities of short-context VLMs to long-context comprehension, without relying on any annotated long-video data. Compared with specialized long-video models, LongVPO achieves the state-of-the-art on both long and short video understanding benchmarks. These advances highlight its potential as a general-purpose solution for long video understanding.

Limitations. Our work prioritizes performance improvement over inference computational efficiency. We will explore the integration with existing context compression approaches in future research.

Acknowledgement

This work is supported by the National Key R&D Program of China (No. 2022ZD0160900), the Natural Science Foundation of Jiangsu Province (No. BK20250009), Nanjing University-China Mobile Communications Group Co., Ltd. Joint Institute (No. NJ20250033), and the Collaborative Innovation Center of Novel Software Technology and Industrialization.

Zhenpeng Huang and Jiaqi Li contribute equally to this work.

References

[1] 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] Daechul Ahn, Yura Choi, San Kim, Youngjae Yu, Dongyeop Kang, and Jonghyun Choi. Isr-dpo: Aligning large multimodal models for videos by iterative self-retrospective dpo. *arXiv preprint arXiv:2406.11280v2*, 2025.

[3] Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, Jian-Guang Lou, and Weizhu Chen. Make your llm fully utilize the context. *Advances in Neural Information Processing Systems*, 37:62160–62188, 2024.

[4] Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, et al. Internlm2 technical report. *arXiv preprint arXiv:2403.17297*, 2024.

[5] Lin Chen, Xilin Wei, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Bin Lin, Zhenyu Tang, Li Yuan, Yu Qiao, Dahua Lin, Feng Zhao, and Jiaqi Wang. Sharegpt4video: Improving video understanding and generation with better captions. *arXiv preprint arXiv:2406.04325*, 2024.

[6] Yukang Chen, Fuzhao Xue, Dacheng Li, Qinghao Hu, Ligeng Zhu, Xiuyu Li, Yunhao Fang, Haotian Tang, Shang Yang, Zhijian Liu, Yihui He, Hongxu Yin, Pavlo Molchanov, Jan Kautz, Linxi Fan, Yuke Zhu, Yao Lu, and Song Han. LongVILA: Scaling long-context visual language models for long videos. In *The Thirteenth International Conference on Learning Representations*, 2025.

[7] Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.

[8] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198, 2024.

[9] Chuanqi Cheng, Jian Guan, Wei Wu, and Rui Yan. Scaling video-language models to 10k frames via hierarchical differential distillation. *arXiv preprint arXiv:2504.02438*, 2025.

[10] 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.

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

[12] Jiajun Fei, Dian Li, Zhidong Deng, Zekun Wang, Gang Liu, and Hui Wang. Video-ccam: Enhancing video-language understanding with causal cross-attention masks for short and long videos, 2024.

[13] Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Caifeng Shan, Ran He, and Xing Sun. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24108–24118, 2025.

[14] Jinlan Fu, Shenzhen Huangfu, Hao Fei, Xiaoyu Shen, Bryan Hooi, Xipeng Qiu, and See-Kiong Ng. Chip: Cross-modal hierarchical direct preference optimization for multimodal llms. *arXiv preprint arXiv:2501.16629*, 2025.

[15] Haojian Huang, Haodong Chen, Shengqiong Wu, Meng Luo, Jinlan Fu, Xinya Du, Hanwang Zhang, and Hao Fei. Vistadpo: Video hierarchical spatial-temporal direct preference optimization for large video models. *arXiv preprint arXiv:2504.13122*, 2025.

[16] Zhenpeng Huang, Xinhao Li, Jiaqi Li, Jing Wang, Xiangyu Zeng, Cheng Liang, Tao Wu, Xi Chen, Liang Li, and Limin Wang. Online video understanding: Ovbench and videochat-online. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3328–3338, 2025.

[17] Yogesh Kulkarni and Pooyan Fazli. Videopasta: 7k preference pairs that matter for video-llm alignment. *arXiv preprint arXiv:2504.14096*, 2025.

[18] Li KunChang, He Yinan, Wang Yi, Li Yizhuo, Wang Wenhai, Luo Ping, Wang Yali, Wang Limin, and Qiao Yu. Videochat: Chat-centric video understanding. *arXiv preprint arXiv:2305.06355*, 2023.

[19] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer, 2024.

[20] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, Limin Wang, and Yu Qiao. 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.

[21] Pengyi Li, Irina Abdullaeva, Alexander Gambashidze, Andrey Kuznetsov, and Ivan Oseledets. Maxinfo: A training-free key-frame selection method using maximum volume for enhanced video understanding. *arXiv preprint arXiv:2502.03183*, 2025.

[22] Rui Li, Xiaohan Wang, Yuhui Zhang, Zeyu Wang, and Serena Yeung-Levy. Temporal preference optimization for long-form video understanding. *arXiv preprint arXiv:2501.13919*, 2025.

[23] Shicheng Li, Lei Li, Kun Ouyang, Shuhuai Ren, Yuanxin Liu, Yuanxing Zhang, Fuzheng Zhang, Lingpeng Kong, Qi Liu, and Xu Sun. Temple:temporal preference learning of video llms via difficulty scheduling and pre-sft alignment. *arXiv preprint arXiv:2503.16929*, 2025.

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

[25] Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 2023.

[26] Jiajun Liu, Yibing Wang, Hanghang Ma, Xiaoping Wu, Xiaoqi Ma, Xiaoming Wei, Jianbin Jiao, Enhua Wu, and Jie Hu. Kangaroo: A powerful video-language model supporting long-context video input, 2024.

[27] Ziyu Liu, Yuhang Zang, Xiaoyi Dong, Pan Zhang, Yuhang Cao, Haodong Duan, Conghui He, Yuanjun Xiong, Dahua Lin, and Jiaqi Wang. Mia-dpo: Multi-image augmented direct preference optimization for large vision-language models. *arXiv preprint arXiv:2410.17637*, 2024.

[28] Zuyan Liu, Yuhao Dong, Ziwei Liu, Winston Hu, Jiwen Lu, and Yongming Rao. Oryx MLLM: On-demand spatial-temporal understanding at arbitrary resolution. In *The Thirteenth International Conference on Learning Representations*, 2025.

[29] Zhijian Liu, Ligeng Zhu, Baifeng Shi, Zhuoyang Zhang, Yuming Lou, Shang Yang, Haocheng Xi, Shiyi Cao, Yuxian Gu, Dacheng Li, Xiuyu Li, Haotian Tang, Yunhao Fang, Yukang Chen, Cheng-Yu Hsieh, De-An Huang, An-Chieh Cheng, Jinyi Hu, Sifei Liu, Ranjay Krishna, Pavlo Molchanov, Jan Kautz, Hongxu Yin, Song Han, and Yao Lu. Nvila: Efficient frontier visual language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4122–4134, 2025.

[30] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz 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*, 2024.

[31] OpenAI. Hello gpt-4o. In *OpenAI Blog*, 2024.

[32] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. Featured Certification.

[33] Richard Yuanzhe Pang, Weizhe Yuan, Kyunghyun Cho, He He, Sainbayar Sukhbaatar, and Jason Weston. Iterative reasoning preference optimization. In *Advances in Neural Information Processing Systems*, pages 116617–116637. Curran Associates, Inc., 2024.

[34] Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. YaRN: Efficient context window extension of large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[35] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741, 2023.

[36] Ruchit Rawal, Khalid Saifullah, Miquel Farré, Ronen Basri, David Jacobs, Gowthami Somepalli, and Tom Goldstein. Cinepile: A long video question answering dataset and benchmark. *arXiv preprint arXiv:2405.08813*, 2024.

[37] Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

[38] Xiaoqian Shen, Yunyang 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. In *Forty-second International Conference on Machine Learning*, 2025.

[39] Yunhang Shen, Chaoyou Fu, Shaoqi Dong, Xiong Wang, Yi-Fan Zhang, Peixian Chen, Mengdan Zhang, Haoyu Cao, Ke Li, Xiawu Zheng, Yan Zhang, Yiyi Zhou, Ran He, Caifeng Shan, Rongrong Ji, and Xing Sun. Long-vita: Scaling large multi-modal models to 1 million tokens with leading short-context accuracy. *arXiv preprint arXiv:2502.05177*, 2025.

[40] Yan Shu, Zheng Liu, Peitian Zhang, Minghao Qin, Junjie Zhou, Zhengyang Liang, Tiejun Huang, and Bo Zhao. Video-xl: Extra-long vision language model for hour-scale video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26160–26169, 2025.

[41] Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, et al. Moviechat: From dense token to sparse memory for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18221–18232, 2024.

[42] Weihan Wang, Zehai He, Wenyi Hong, Yean Cheng, Xiaohan Zhang, Ji Qi, Xiaotao Gu, Shiyu Huang, Bin Xu, Yuxiao Dong, et al. Lvbench: An extreme long video understanding benchmark. *arXiv preprint arXiv:2406.08035*, 2024.

[43] Yi Wang, Xinhao Li, Ziang Yan, Yinan He, Jiashuo Yu, Xiangyu Zeng, Chenting Wang, Changlian Ma, Haian Huang, Jianfei Gao, et al. Internvideo2. 5: Empowering video mllms with long and rich context modeling. *arXiv preprint arXiv:2501.12386*, 2025.

[44] Hongchen Wei, Zhihong Tan, Yaosi Hu, Chang Wen Chen, and Zhenzhong Chen. Longcaptioning: Unlocking the power of long video caption generation in large multimodal models. *arXiv preprint arXiv:2502.15393*, 2025.

[45] Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. Longvideobench: A benchmark for long-context interleaved video-language understanding. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

[46] Yuxi Xie, Guanzhen Li, Xiao Xu, and Min-Yen Kan. V-dpo: Mitigating hallucination in large vision language models via vision-guided direct preference optimization. *arXiv preprint arXiv:2411.02712*, 2024.

[47] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[48] Dongjie Yang, Suyuan Huang, Chengqiang Lu, Xiaodong Han, Haoxin Zhang, Yan Gao, Yao Hu, and Hai Zhao. Vript: A video is worth thousands of words. *Advances in Neural Information Processing Systems*, 37:57240–57261, 2024.

[49] 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.

[50] 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. In *The Thirteenth International Conference on Learning Representations*, 2025.

[51] 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.

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

[53] Hang Zhang, Xin Li, and Lidong Bing. Video-LLaMA: An instruction-tuned audio-visual language model for video understanding. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2023.

[54] 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.

[55] Ruohong Zhang, Liangke Gui, Zhiqing Sun, Yihao Feng, Keyang Xu, Yuanhan Zhang, Di Fu, Chunyuan Li, Alexander Hauptmann, Yonatan Bisk, et al. Direct preference optimization of video large multimodal models from language model reward. *arXiv preprint arXiv:2404.01258*, 2024.

[56] Shaolei Zhang, Qingkai Fang, Zhe Yang, and Yang Feng. LLaVA-mini: Efficient image and video large multimodal models with one vision token. In *The Thirteenth International Conference on Learning Representations*, 2025.

[57] 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.

[58] Zhiyuan Zhao, Bin Wang, Linke Ouyang, Xiaoyi Dong, Jiaqi Wang, and Conghui He. Beyond hallucinations: Enhancing lvlms through hallucination-aware direct preference optimization. *arXiv preprint arXiv:2311.16839*, 2023.

[59] Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Zhengyang Liang, Shitao Xiao, Minghao Qin, Xi Yang, Yongping Xiong, Bo Zhang, Tiejun Huang, and Zheng Liu. Mlvu: Benchmarking multi-task long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13691–13701, 2025.

[60] Yiyang Zhou, Chenhang Cui, Rafael Rafailov, Chelsea Finn, and Huaxiu Yao. Aligning modalities in vision large language models via preference fine-tuning. *arXiv preprint arXiv:2402.11411*, 2024.

Appendix Overview

This appendix provides additional details of our approach and further experimental results, organized as follows:

- Section A describes the core experimental design behind our context-bias analysis.
- Section B showcases qualitative results across various scenarios.
- Section C outlines implementation details of our method.

Model	LVBench	LongVideoBench	MLVU	Video-MME
Qwen2.5-VL	45.3	56.0	70.2	65.1 / 71.6
InternVL2.5	45.2	62.7	67.6	61.1 / 65.3
+ LongVPO	**50.1** ↑4.9	**66.6** ↑3.9	**74.1** ↑6.5	64.6 / 70.3 ↑3.5/↑5.0
InternVideo2.5	47.4	63.2	72.8	63.3 / 71.1
+ LongVPO	**51.0** ↑3.6	**67.2** ↑4.0	**74.7** ↑1.9	66.1 / 73.1 ↑2.8/↑2.0
Video-LLaMA3	45.3	59.8	73.0	66.2 / 70.3
+ LongVPO	**49.8** ↑4.5	**63.4** ↑3.6	**74.6** ↑1.6	67.2 / 71.4 ↑1.0/↑1.3

Table 3: Performance comparison on long video benchmarks. Improvements over base models are shown in red with ↑ symbols.

A Core Experimental Design for Context Position Bias Probing (Main Fig. 1)

Evaluation Setup. We directly selected tasks from MVBench [20] for evaluation. Only unambiguous tasks were included to ensure the validity of the labels.

To evaluate models designed primarily for short-context input, we introduce a simplified evaluation framework: (1) **Padding Strategy:** We simulate long-context scenarios by embedding the original video frame sequence into a larger grid (akin to high-resolution image tiling), surrounded by meaningless padding frames. (2) **Random Placement:** The original frames are randomly placed within this padded grid to test whether a model's performance is sensitive to the spatial location of meaningful content relative to the padding.

This experiment aims to validate two key constraints for an ideal long-video context model: (1) **Consistency across Context Lengths:** A well-designed model should maintain consistent performance across both short and long contexts without altering the task semantics. (2) **Position Invariance:** Since padding frames carry no meaningful information, the spatial location of valid video frames within the padded grid should not affect task performance.

As shown in Main Fig. 1, existing long-context models fall short of these expectations: (a) **Shifted Long-Context Consistency:** Performance varies with the distance between query and relevant frames, showing an undesirable sensitivity to position. An ideal long-video model should attend equally to relevant frames regardless of their location in the input. (b) **Short-Long Context Discrepancy:** Compared to our LongVPO, existing models exhibit a significant performance drop when transitioning from short to long context inputs. This suggests unreliable long-video understanding. When used directly as the reference model in DPO-style fine-tuning with long-context inputs, these models may lead to suboptimal performance. In contrast, LongVPO maintains nearly identical performance across short and long contexts, validating its robust design.

B Qualitative Results

We present additional qualitative comparisons with state-of-the-art models on long-video tasks across diverse scenarios. Despite being trained on synthetic data, our model demonstrates competitive open-ended QA performance, maintaining robustness across various domains.

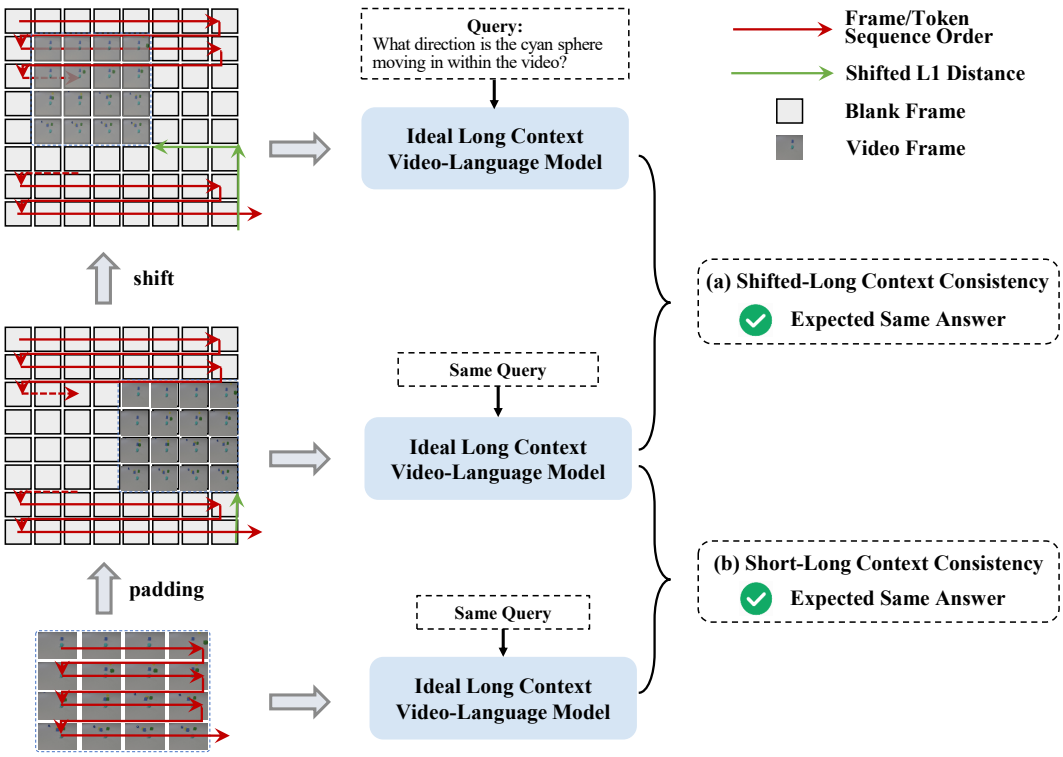


Figure 8: Core experimental design underpinning the context-bias analysis in Fig. 1.

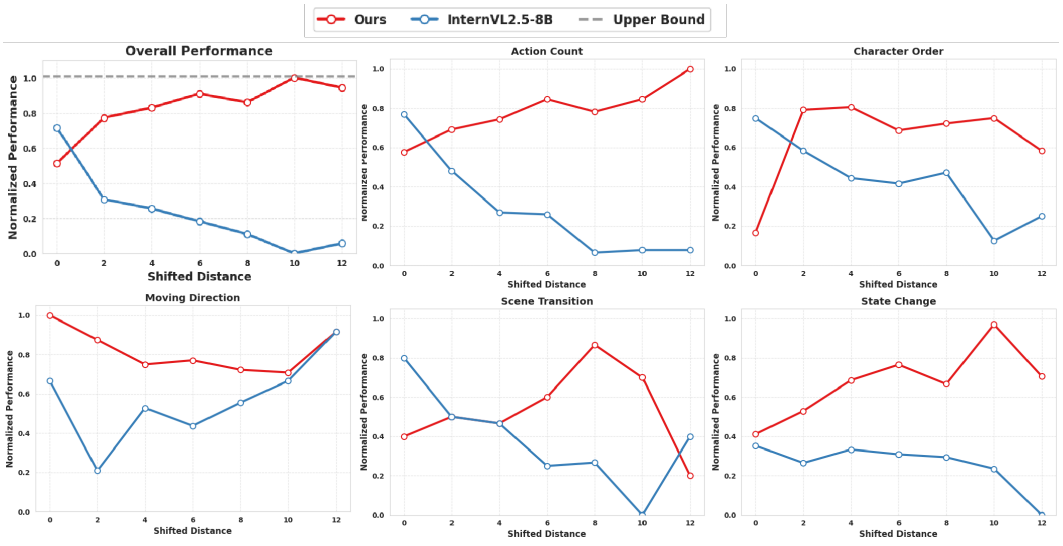


Figure 9: Compared to the main Fig. 1, we shorten the input video context length (by padding blank frames to a 10×10 grid rather than 12×12), yet the same "lost in the middle" phenomenon persists.

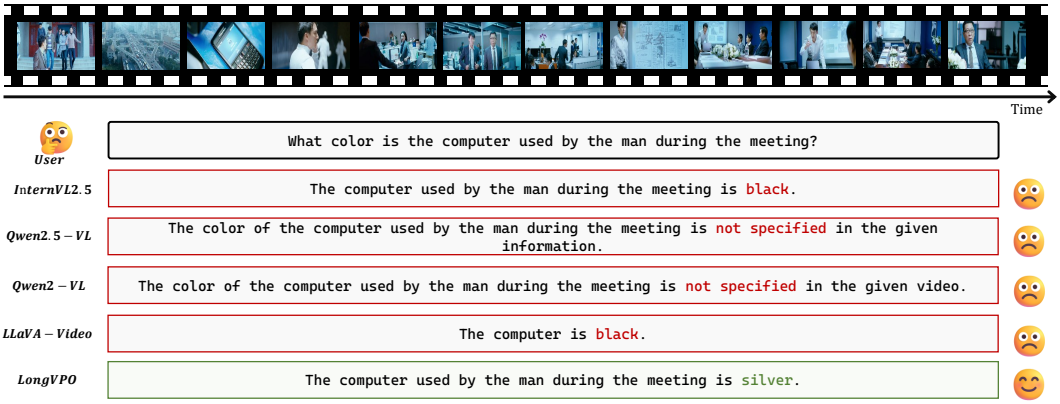


Figure 10: Long Video Understanding: Visual Semantic Understanding

Figure 11: Long Video Understanding: Cross-temporal Scene Association

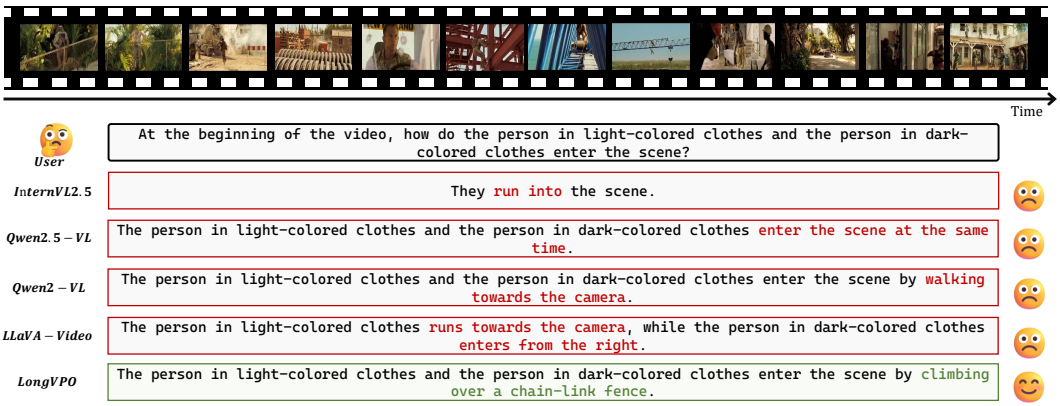


Figure 12: Long Video Understanding: Temporal Order Analysis

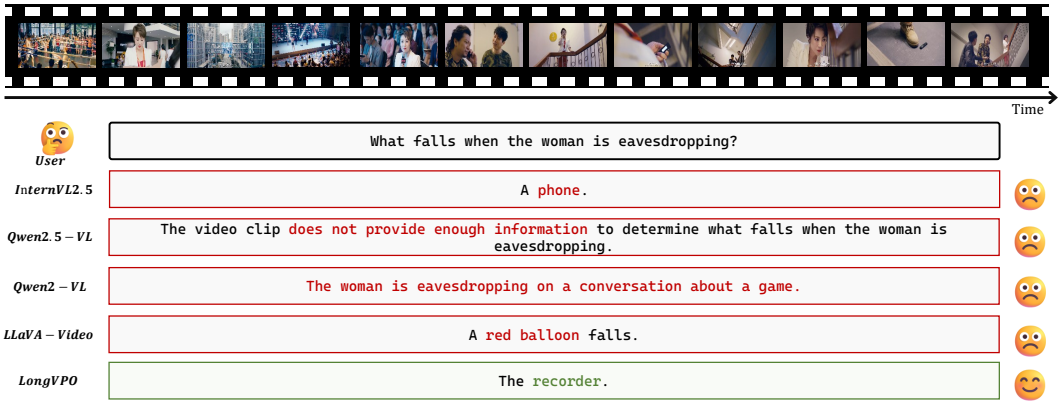


Figure 13: Long Video Understanding: Detail Comprehension

C Implementation Details

Hardware Setup. All experiments were conducted on a server equipped with 4×8 NVIDIA H100 GPUs, each with 80GB of memory. We implement DeepSpeed Ulysses sequence parallelism to enable memory-efficient training.

Training Strategy. We adopt the proposed LongVPO method for training, which involves a two-stage fine-tuning process on a curated dataset of 16k samples—10k samples for stage 1 and 6k for stage 2. Each model variant is trained for 1 epoch, balancing training efficiency with the need for robust adaptation.

Full-Model Fine-Tuning. We fine-tune the entire model end-to-end. This includes the vision encoder, the vision-language connector, and the LLM backbone.

Optimization Settings. We use a composite loss that balances KL-divergence and supervised fine-tuning (SFT) objectives, with both weights set to 1.0 (β=0.01, α=1.0). The learning rate is set to 5e-7, with batch size 8, a cosine learning rate scheduler, and a warm-up ratio of 0.01 to stabilize early training dynamics.

Training Duration. Each model variant requires approximately 10 hours of training with DeepSpeed Ulysses sequence parallelism enabled; otherwise, training completes in about 1 hour under the aforementioned configuration.

Evaluation Settings. For consistency and comparability, we follow the evaluation protocols established by InternVL2.5 and InternVideo2.5. The maximum number of frames per input is set to 512 to test the model's scalability in long-context understanding.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In the abstract and introduction sections, we have outlined the contributions of our work and provided a summary at the end of the introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: In the conclusion section, we discuss the limitations of our work and suggest directions for future improvements.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: Our work does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: In the methods section, we provide a detailed description of the training methods. Additionally, the implementation details are discussed in the experiments and appendix sections.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Data and code will be available upon acceptance.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: At the beginning of the experiments section and in the appendix, we detail the implementation specifics for each experiment.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Due to limited computational resources, our paper does not include error bars or extensive statistical significance analysis for the experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: In the appendix, we provide detailed information regarding the computational resources utilized for each experiment.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We have thoroughly reviewed and adhered to the NeurIPS Code of Ethics throughout our research process.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

    Justification: Our work focuses on advancing long video understanding with multimodal large models. It provides technical improvements in training method and training efficiency without introducing new capabilities or applications that would create additional societal impacts.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: There is no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Yes, we have properly credited the creators or original owners of assets used in our paper, and we have explicitly mentioned and respected the license and terms of use associated with them.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Corresponding assets will be available upon acceptance.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Our work does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our work does not involve crowdsourcing nor research with human subjects

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.